Manual de Programação

TPW 03

Controlador Programável

CAPÍTULO 1
INSTRUÇÕES BÁSICAS DE PROGRAMAÇÃO 1

CAPÍTULO 2
DETALHAMENTO DOS ENDEREÇOS DE MEMÓRIA

CAPÍTULO 3
INSTRUÇÕES APLICÁVEIS

CAPÍTULO 4
MARCADORES ESPECIAIS

CAPÍTULO 5
TABELAS DE ENDEREÇO DO TPW-03

CAPÍTULO 6
OP07/OP08 INSTRUÇÃO DE OPERAÇÃO

CAPÍTULO 1

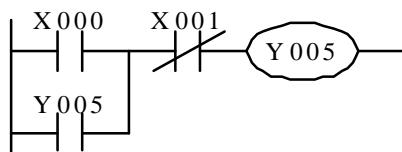
INSTRUÇÕES BÁSICAS DE PROGRAMAÇÃO 1

1 INSTRUÇÕES BÁSICAS DE PROGRAMAÇÃO

1.1 O QUE É UM PROGRAMA?

Um programa é uma série de instruções conectadas escritas numa linguagem que um PLC pode compreender. Existem dois tipos de formato de programa: lista de instruções e ladder.

0	LD	X000
1	OR	Y005
2	ANI	X001
3	OUT	Y005

X000 X001 Y005
Y005

Lista de Instruções   Ladder

1.2 ENDEREÇOS BÁSICOS DE MEMÓRIA USADOS NA PROGRAMAÇÃO

Basicamente existem seis tipos de endereços de memória que podem ser utilizados para programação do PLC. Cada endereço tem sua característica específica. Para viabilizar a rápida e fácil identificação de tipo de memória, cada endereço possui uma letra inicial específica;

- X: Identifica todas as entradas físicas e diretas ao PLC.
- Y: Identifica todas as saídas físicas e diretas do PLC.
- T: Identifica os temporizadores internos do PLC.
- C: Identificar os contadores internos do PLC.
- M e S: São marcadores auxiliares que podem ser utilizados para nas operações internas do programa do PLC.

Todos os endereços mencionados acima são endereços de BIT, ou seja, podem armazenar somente dois estados, ON ou OFF, ou ainda 1 ou 0.

Informações detalhadas dos endereços de memória:

☑ O Capítulo 2 apresenta estas informações de forma detalhada. Entretanto, o acima exposto é todo o necessário para o restante deste capítulo.

1.3 COMO INTERPRETAR A LINGUAGEM *LADDER*

A linguagem *Ladder* é muito semelhante á lógica de relés utilizada em circuitos elétricos. Existem tanto contatos quanto bobinas que podem ser de diversas formas para elaboração do programa. Entretanto, o princípio básico permanece o mesmo. Uma bobina aciona saídas diretas do PLC (ex. um endreço Y) ou aciona temporizadores internos, contadores ou ainda marcadores auxiliares (ex. endereços T, C, M e S). Cada bobina têm contatos auxiliares que ser utilizados em outros locais na lógica. Estes contatos estão disponíveis tanto nas configurações "normalmente aberto" (NA) quanto "normalmente fechado" (NF).

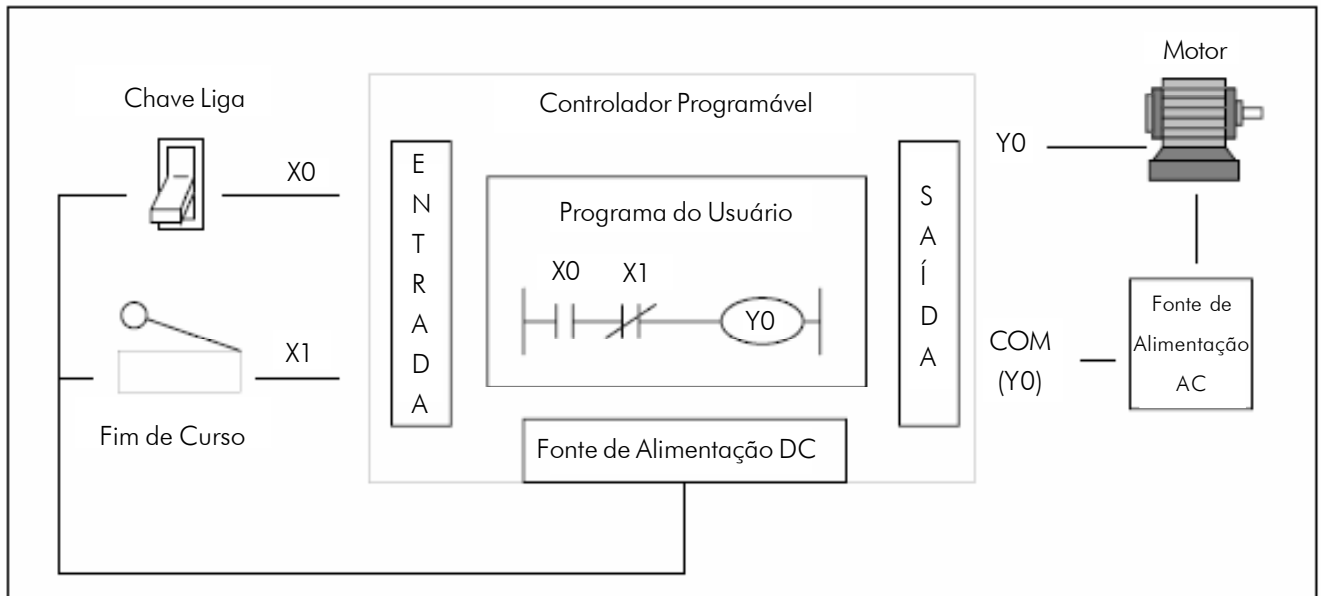
O termo “normal(mente)” refere-se ao estado dos contatos quando a bobina não está energizada.

Usando uma analogia de relé, quando a bobina está OFF, um contato NA não estaria conduzindo corrente, isto é, uma carga sendo acionada através de um contato NA não estaria ligada. Entretanto, um contato NF permitiria que existisse fluxo de corrente, portanto a carga conectada estaria ativa.

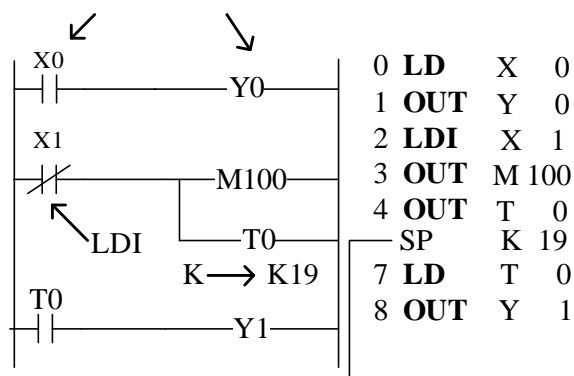
Ativar a bobina reverte o estado do contato, isto é, a corrente seria conduzida no contato NA e inibida no contato NF.

Entradas físicas ao PLC (endereços X) não podem ser utilizados como bobinas de saída programáveis. Estes endereços só podem ser usados num formato de contato (tipos NA e NF estão disponíveis).

Exemplo:

Justamente por causa da associação da lógica LADDER aos circuitos elétricos, o programa pode ser interpretado sempre da esquerda para a direita, ou seja, a corrente deve passar por diversos contatos, por exemplo, do tipo X0 e X1, para ligar a bobina de saída Y0 na posição ON. Portanto, no exemplo mostrado, ligando o X0 na posição ON faz com que a saída Y0 também ligue na posição ON. Se, no entanto, a chave limite X1 é ativada, a saída Y0 desliga na posição OFF. Isto porque o fluxo do sinal deveria passar por X0 e X1 para acionar a saída, mas como X1 em ON bloqueia do fluxo do sinal a saída é desenergizada.

1.4 INÍCIO DE LÓGICA (LOAD, LOAD INVERSE)

Mnemônico	Função	Formato	Tipo de endereço	Passo
[LD]	Contato inicial de lógica do tipo NA (normalmente aberto)	XYMSTC	X,Y,M,S,T,C	1
[LDI]	Contato inicial de lógica do tipo NF (normalmente fechado)	XYMSTC	X,Y,M,S,T,C	1

Exemplo de Programa:

Pontos básicos para observar:

- As instruções LD e LDI devem sempre ser utilizadas no início de cada ramo de lógica, conectada a linha esquerda do circuito.
- LD e LDI também podem ser utilizadas para informar o início de uma seqüência lógica dentro do ramo quando estiver usando as instruções ORB e ANB (ver as seções seguintes).

A instrução OUT:

☑ Para detalhes da instrução OUT (incluindo as variações do temporizador básico e contador) ver os próximos itens.

1.5 INSTRUÇÃO DE SAÍDA (OUT)

Mnemônico	Função	Formato	Tipos de endereços	Passos do programa
[OUT]	Finaliza um ramo lógico. Armazena o status da lógica.	YMSTC	Y,M,S,T,C	Y,M: 1 S, special M Bobina: 2 T: 3 C (16bits): 3 C (32bits): 5

Pontos básicos para observar:

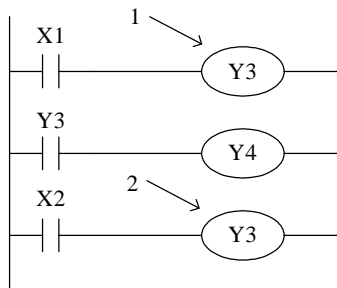
- A instrução OUT sempre deve ser conectada a linha direita do circuito para finalizar um ramo de lógica.
- Os endereços de entrada do tipo 'X' não podem ser utilizados na instrução OUT.

É possível conectar múltiplas instruções OUT em paralelo (ver o exemplo na página anterior: configuração M100/T0).

1.5.1 TIPOS DE TEMPORIZADORES E CONTADORES

É possível configurar uma instrução OUT como um temporizador (T) ou contador (C), ou até mesmo como constantes. A constante é identificada pela letra "K" (veja o exemplo na página anterior: T0 K19). No caso de um temporizador, a constante "K" define o tempo que a temporização irá durar até energizar a bobina de saída uma vez que a entrada do temporizar esteja habilitada, por exemplo, se um temporizador com base de tempo de 100 mseg tem uma constante "K100", ele levará (100x100 msec) 10 segundos para energizar a saída do temporizador uma vez que sua entrada é habilitada. Com contadores, as constantes identificam quantas vezes a entrada do contador deverá receber um pulso para efetivamente ativar a saída, por exemplo, o contador com um constante "8" deverá receber 8 pulsos em sua entrada antes que sua bobina de saída seja energizada. A tabela abaixo identifica alguns parâmetros básicos para os vários temporizadores e contadores;

Temporizador/Contador	Programação da constante K	Valor real de temporização/ contagem	Passos do programa
Temporizador com base de tempo de 1ms	1~32.767	0,001~32,676 seg	3
Temporizador com base de tempo de 10ms	1~32.767	0,01~327,67 seg	
Temporizador com base de tempo de 100ms		0,1~3.276,7 seg	
Contador de 16 bits	1~32.767	1~32.767	
Contador de 32 bits	-2.147.483.648~ +2.147.483.647	-2.147.483.648~ +2.147.483.647	5

1.5.2 UTILIZANDO A MESMA MEMÓRIA EM MAIS DE UMA SAÍDA NO PROGRAMA

Utilizar o mesmo endereço de memória em mais de uma instrução de saída no programa não é uma prática recomendada. Usar múltiplas instruções de saída com o mesmo endereço pode fazer com que o programa não se comporte como o esperado. O exemplo de programa mostrado ao lado mostra uma situação onde um mesmo endereço de memória é utilizando no mesmo programa em mais de uma instrução de saída; por exemplo, existem duas saídas Y3. A seguinte seqüência de eventos ocorrerá quando a entrada X1 = ON e X2 = OFF;

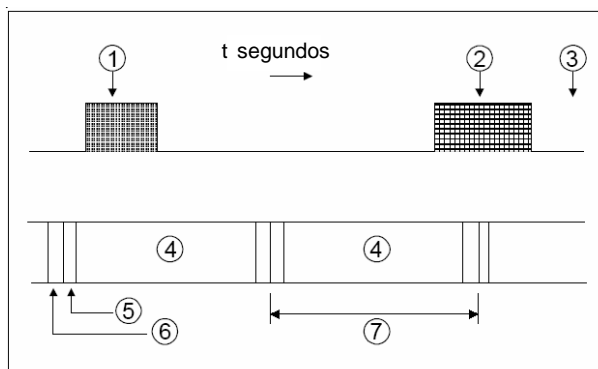
1. A primeira linha lógica de Y3 terá o este endereço com o status ON porque X1 está ON. Os contatos associados com a Y3 também energizarão quando a bobina de saída Y3 energizar. Portanto, a saída Y4 também será energizada (ON).

2. A última e mais importante linha deste programa mostra o estado da entrada X2. Como X2 está desenergizada (OFF), a saída Y3 NÃO será energizada. Portanto, ao final do ciclo de varredura, estado da bobina Y3 que o programa irá mostrar fisicamente será OFF. Os status finais das saídas serão então Y3 = OFF e Y4 = ON.

Uso do mesmo endereço em diversas saídas:

☑ Sempre verifique os programas para evitar situações onde o mesmo endereço de memória tenha sido utilizado mais que uma vez numa instrução de saída do programa. Se houverem duas saídas ou mais com o mesmo endereço, o programa poderá apresentar comportamentos inesperados na saída física do PLC.

O efeito da última bobina:

☑ Quando da repetição de um mesmo endereço em diversas saídas do programa, sempre a última instrução de saída com este endereço é que irá atualizar o estado da saída física.

Durações de entrada:

As durações de ON ou OFF das entradas do PLC devem ser mais longas que o tempo do ciclo de operação do PLC. Levando em consideração um atraso de resposta de 10 mseg (filtro de entrada padrão), a duração de ON/OFF deve ser mais longa que 20 mseg se o ciclo de operação (tempo de scan é 10 mseg. Portanto, neste exemplo, os pulsos de entrada de mais de 25Hz (1seg/(20mseg ON + 20mseg OFF)) não podem ser percebidos.

Existem instruções especiais para tais entradas de alta velocidade.

No desenho acima temos as seguintes condições:

①→Estado ON da entrada NÃO reconhecido

②→Estado ON da entrada reconhecido

③→Estado OFF da entrada NÃO reconhecido

④→Execução do programa

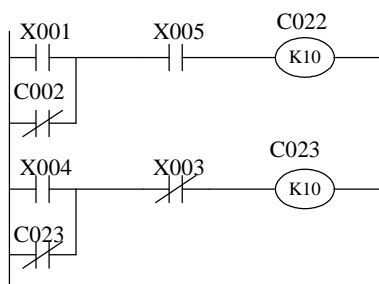
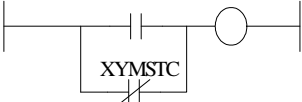
⑤→Atualizando as entradas físicas do programa

⑥→Atualizando as saídas físicas do programa

⑦→Um ciclo de varredura do programa completo

1.6 PORTA LÓGICA "E" E "E" BARRADO (AND, AND NOT)

Mnemônico	Função	Formato	Tipos de endereços	Passos do programa
[AND]	Conexão em série de contatos NA (normalmente aberto)	XYMSTC	X,Y,M,S,T,C	1
[ANI]	Conexão em série de contatos NF (normalmente fechado)	XYMSTC	X,Y,M,S,T,C	1

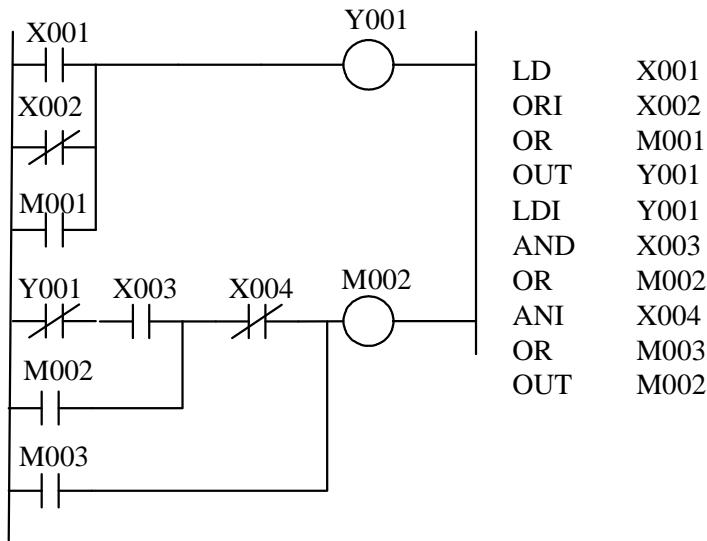
Exemplo de programa:

```
LD    X001
ORI   C002
AND   X005
OUT   C022  K10

LD    X004
ORI   C003
ANI   X003
OUT   C023  K10
```

Pontos básicos para lembrar:

- Use as instruções AND e ANI para conexão em série de contatos. Podem ser conectados em série tantos contatos quanto requeridos (o número de contatos em série não é limitado).

1.7 PORTA LÓGICA "OU" E "OU" BARRADO (OR, OR NOT)

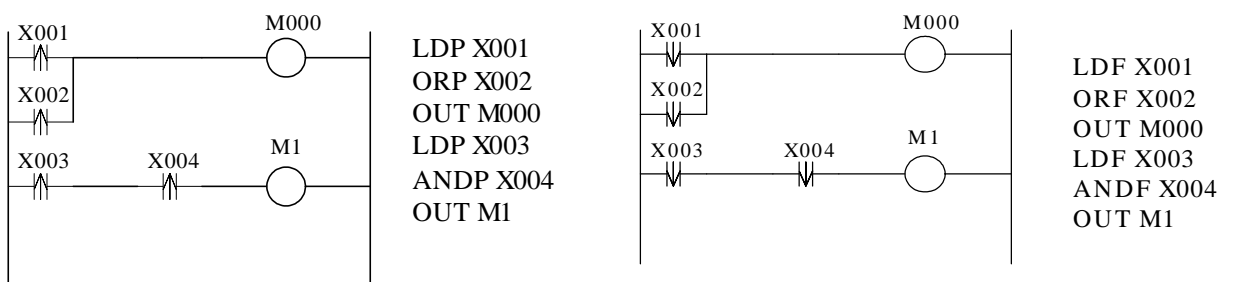
Mnemônico	Função	Formato	Tipos de endereços	Passos do programa
[OR]	Conexão paralela de contatos NA (normalmente aberto)	XYMSTC	X,Y,M,S,T,C	1
[ORI]	Conexão paralela de contatos NF (normalmente fechado)	XYMSTC	X,Y,M,S,T,C	1

Exemplo de programa:

Pontos básicos para lembrar:

- Use as instruções OR e ORI para conexão paralela dos contatos. Para conectar um bloco que contem mais do que um contato conectado em série a outro bloco de circuito em paralelo, use uma instrução ORB.
- Conecte um lado da instrução OR/ORI a linha da esquerda do circuito lógico.

1.8 PULSO DA BORDA DE SUBIDA, PULSO NA BORDA DE DESCIDA (INÍCIO DE LÓGICA)

Mnemônico	Função	Formato	Tipos de endereços	Passos do programa
[LDP]	Operação lógica inicial – Pulso da borda de subida	XYMSTC	X,Y,M,S,T,C	2
[LDF]	Operação lógica inicial - Pulso da borda de descida	XYMSTC	X,Y,M,S,T,C	2

Exemplo de programa:

Pontos básicos para lembrar:

- Conecte as instruções LDP e LDF diretamente a linha da esquerda do circuito lógico.
- Ou use as instruções LDP e LDF para definir um novo bloco de programa quando usar as instruções ORB e ANB (ver seções adiante).
- A saída da instrução LDP ficará ativa durante um ciclo de varredura do programa após o endereço associado mudar de OFF para ON.
- A saída da instrução LDF ficará ativa durante um ciclo de varredura do programa após o endereço associado mudar de ON para OFF.

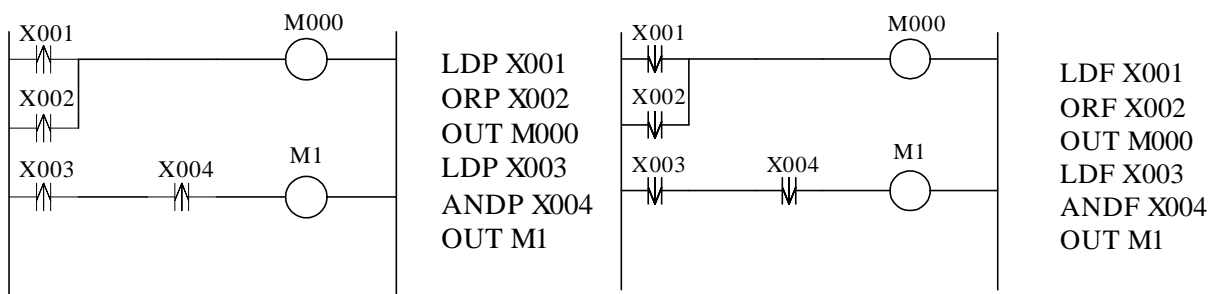
Marcadores de operação única M2800 à M3071:

☑ As instruções de operação de pulso, quando usadas com os marcadores auxiliares M2800 à M3071, só ativam a primeira instrução encontrada na varredura do programa, após o ponto no programa onde o endereço alterar. Quaisquer outras instruções de operação de pulso permanecerão inativas.

☑ Quaisquer outras instruções (LD, AND, OR, etc.) operarão normalmente.

1.9 PULSO DA BORDA DE SUBIDA, PULSO NA BORDA DE DESCIDA (EM SÉRIE)

Mnemônico	Função	Formato	Tipos de endereços	Passos do programa
[ANDP]	Conexão em série do pulso da borda de subida	XYMSTC	X,Y,M,S,T,C	2
[ANDF]	Conexão em série do pulso da borda de descida	XYMSTC	X,Y,M,S,T,C	2

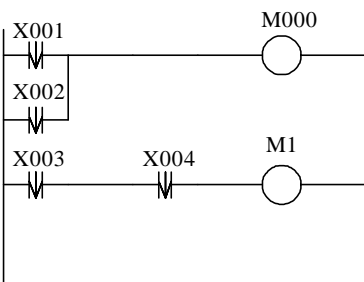
Exemplo do programa:

Pontos básicos para lembrar:

- Use as instruções ANDP e ANDF para a conexão em série do contato de pulso.
- O uso é o mesmo usado para AND e ANI; ver anteriormente.
- A saída da instrução ANDP ficará ativa durante um ciclo de varredura do programa após o endereço associado mudar de OFF para ON.
- A saída da instrução ANDF ficará ativa durante um ciclo de varredura do programa após o endereço associado mudar de ON para OFF.

Marcadores de operação única M2800 à M3071:

Quando usado com os marcadores M2800 à M3071 somente a primeira instrução será ativada. Para detalhes ver página anterior.

1.10 PULSO DA BORDA DE SUBIDA, PULSO NA BORDA DE DESCIDA (EM PARALELO)

Mnemônico	Função	Formato	Tipos de endereços	Passos do programa
[ORP]	Conexão paralela do pulso da borda de subida	XYMSTC	X,Y,M,S,T,C	2
[ORF]	Conexão paralela do pulso da borda de descida	XYMSTC	X,Y,M,S,T,C	2

```
LDF X001
ORF X002
OUT M000
LDF X003
ANDF X004
OUT M1
```

Exemplo de programa:

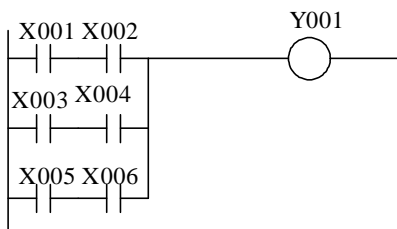
Pontos básicos para lembrar:

- Use as instruções ORP e ORF para conexão paralela dos contatos de pulso.
- A saída da instrução ORP ficará ativa durante um ciclo de varredura do programa após o endereço associado mudar de OFF para ON.
- A saída da instrução ORF ficará ativa durante um ciclo de varredura do programa após o endereço associado mudar de ON para OFF.

Marcadores de operação única M2800 à M3071:

☑ Quando usado com marcadores M2800 à M3071 somente a primeira instrução será ativada. Para detalhes ver páginas anteriores.

1.11 INSTRUÇÃO "OU" PARA CIRCUITOS LÓGICOS

Mnemônico	Função	Formato	Tipos de endereços	Passos do programa
[ORB]	Conexão paralela de múltiplos circuitos de contato	X001 X002 Y001 X003 X004	N/A	1

```
LD X001
AND X002
LD X003
AND X004
ORB
LD X005
AND X006
ORB
OUT Y001
```

Pontos básicos para lembrar:

ma instrução ORB é uma instrução independente e não está associada com nenhum endereço.

- Use a instrução ORB para conectar circuitos lógicos (geralmente blocos de circuito em série) ao circuito paralelo precedente. Blocos de circuito em série são aqueles em que mais de um contato conecta em série ou a instrução ANB é usada.
- Para declarar o ponto de partida do bloco do circuito use uma instrução LD ou LDI.

Após completar o bloco de circuito em série conecte-o ao bloco paralelo precedente usando a instrução ORB.

Limitações de processamento da instrução ORB:

☑ Ao usar as instruções ORB, não use mais do que 8 instruções LD e LDI na definição dos blocos lógicos em paralelo. Ignorar isto resultará em erro de programa.

Limitações de processamento seqüencial:

☑Não há limitações para o número de circuitos paralelos quando usar uma instrução ORB na configuração de processamento seqüencial.

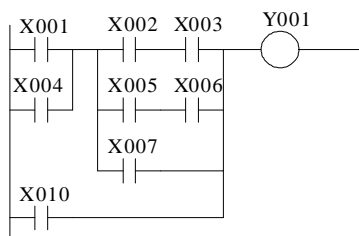
1.12 INSTRUÇÃO "E" PARA CIRCUITOS LÓGICOS

Mnemônico	Função	Formato	Tipos de endereços	Passos do programa
[ANB]	Conexão em série de múltiplos circuitos paralelos	X001 X002 Y001 X003 X004	N/A	1

Exemplo de programa:

Pontos básicos para lembrar:

- Uma instrução ANB é uma instrução independente e não está associada a qualquer endereço.

- Use a instrução ANB para conectar circuitos lógicos (geralmente blocos de circuito paralelos) ao circuito precedente em série. Blocos de circuito paralelo são aqueles nos quais mais do que um contato conecta em paralelo a instrução ORB.

```
LD X001
OR X004
LD X002
AND X003
LDI X005
AND X006
ORB
OR X007
ANB
OR X010
OUT Y001
```

- Para declarar o ponto de partida do bloco do circuito, use uma instrução LD ou LDI. Após completar o bloco de circuito paralelo, conecte-o ao bloco precedente em série usando a instrução ANB.

Limitações de processamento da instrução ANB:

☑ Ao usar as instruções ANB, não use mais do que 8 instruções LD e LDI na definição dos blocos do programa (a serem conectados em paralelo). Ignorar isto resultará em erro de programa.

Limitações de processamento seqüencial:

☑ É possível usar quantas instruções ANB forem necessárias para conectar um número de blocos de circuito paralelo ao bloco precedente em série (ver a lista de programação).

1.13 INSTRUÇÕES MPS, MRD E MPP

Mnemônico	Função	Formato	Tipos de endereços	Passos do programa
[MPS]	Armazena o resultado atual das operações internas do PLC	MPS	N/A	1
[MRD]	Lê o resultado atual das operações internas do PLC	MRD	N/A	1
[MPP]	Remove o resultado atualmente armazenado	MPP	N/A	1

Pontos básicos para lembrar:

- Use estas instruções para conectar as bobinas de saída para o lado esquerdo de um contato. Sem estas instruções conexões só poderão ser feitas do lado direito do último contato.

- A MPS armazena o ponto de conexão do circuito *ladder* para que mais tarde ramificações da bobina possam utilizar este valor.

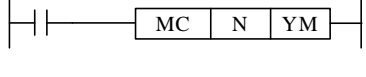
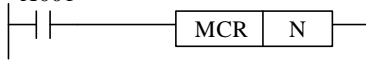
- A MRD lê os dados do ponto de conexão previamente armazenado e força o próximo contato a conectar-se a ele.

- A MPP apaga o ponto de conexão armazenado. Primeiro conecta o próximo contato e depois remove o ponto da área de armazenagem temporária.

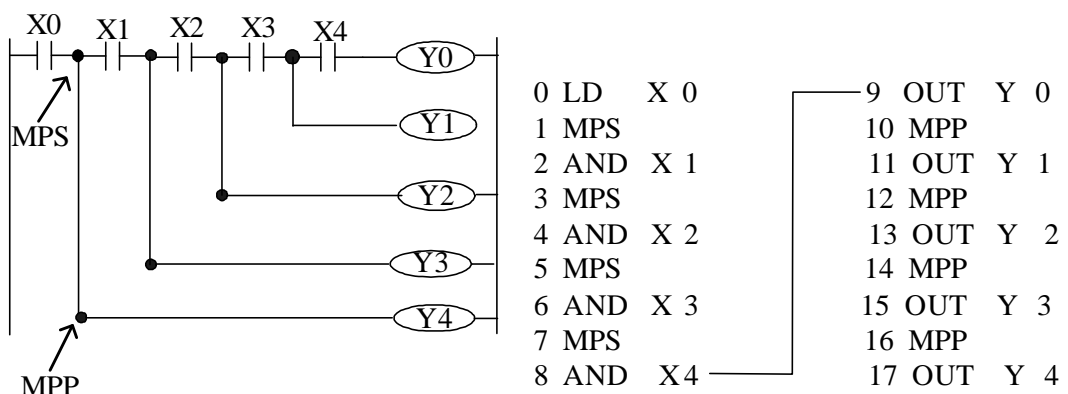
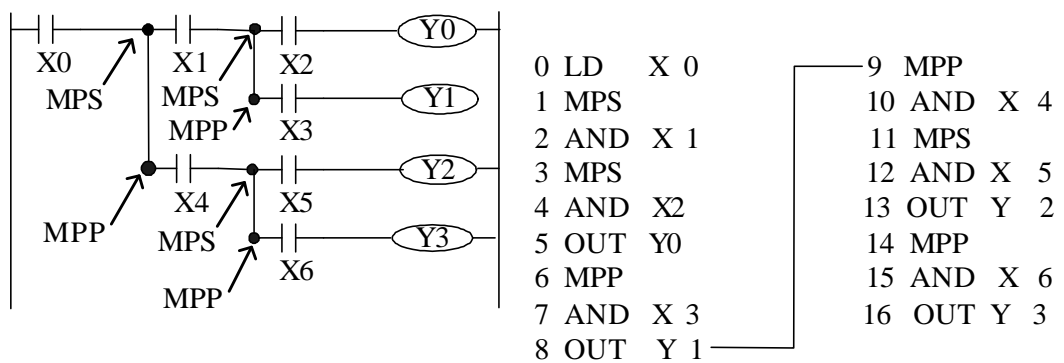
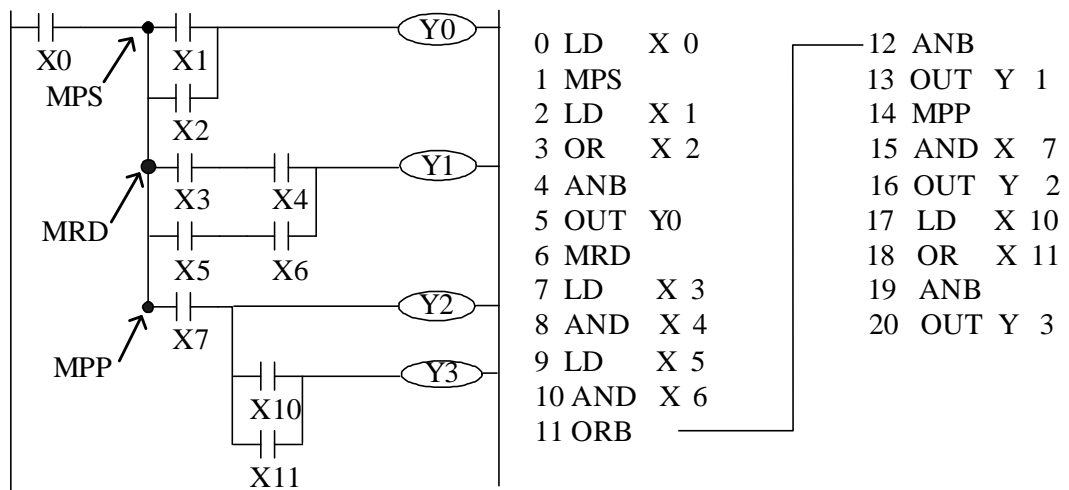
- Para cada instrução MPS DEVE haver uma instrução MPP correspondente.

- O último contato ou circuito de bobina deve conectar numa instrução MPP.

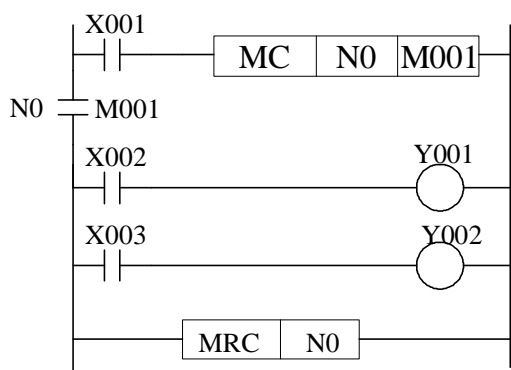
- A qualquer passo da programação, o número de pares MPS-MPP ativos não deve ser maior que 8.

Uso das instruções MPS, MRD e MPP:

☑ Quando estiver editando um programa em linguagem *ladder*, a ferramenta de programação automaticamente adicionará as instruções MPS, MRD e MPP na compilação do programa. Se o programa gerado estiver disponível para visualização, as instruções MPS, MRD e MPP estão presentes.

☑ Quando estiver editando um programa em lista de instruções, é de responsabilidade do programador digitar todas as instruções MPS, MRD e MPP necessárias no programa.

Exemplo de programa:

1.14 SET/RESET DO CONTROLE MESTRE

Mnemônico	Função	Formato	Tipos de endereços	Passos do programa
[MC]	Define o início/partida do controle mestre	X001 — MC N YM	Y,M (nenhuma bobina M especial permitida) N denota o nível do aninhamento (N0 a N7)	3
[MCR]	Define o final do controle mestre	X001 — MCR N	N denota o nível (N0 a N7)a ser zerado	2

Exemplo de programa:

```
LD X001
MC N0
SP M001
LD X002
OUT Y001
LD X003
OUT Y002
MCR N0
```

Pontos básicos para lembrar:

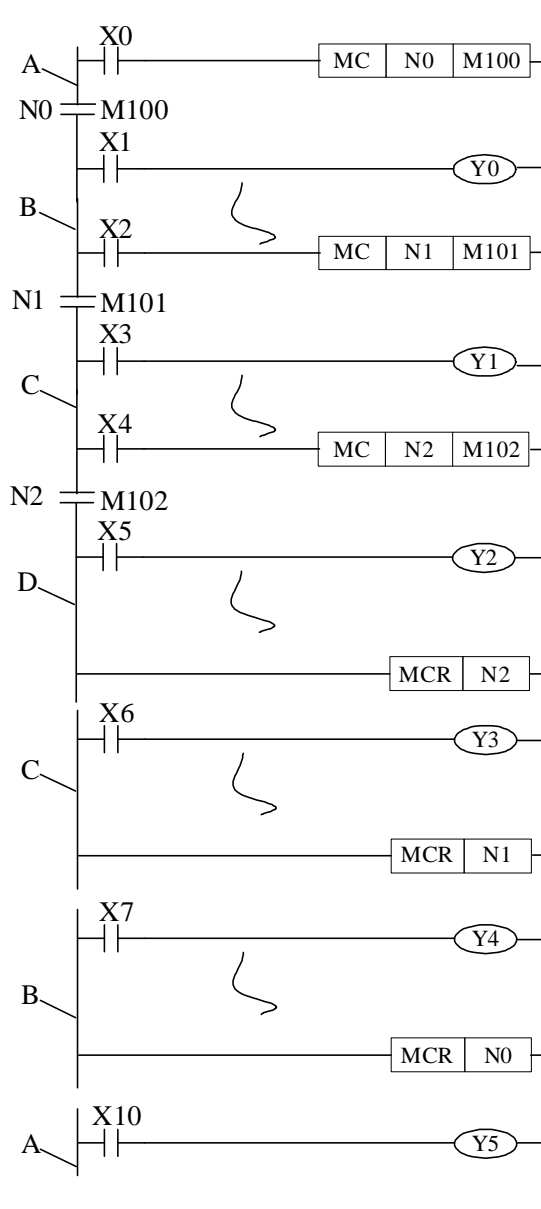
- Após a execução de uma instrução MC, a linha da esquerda do circuito lógico (ponto LD, LDI) mudará para um ponto após a instrução MC. Uma instrução MCR retornará esta condição ao formato original.

- A instrução MC também inclui um ponteiro de nível de aninhamento N. Níveis de aninhamento podem ser definidos num range N0 a N7 (8 pontos). O nível de aninhamento mais elevado é '0' e o menos relevante é '7'.

- A instrução MCR zera cada nível aninhamento. Quando um nível de aninhamento é zerado, ele também zera TODOS os níveis aninhamento menos relevantes. Por exemplo, o MCR N5 reseta níveis de aninhamento de 5 a 7.

- Quando a entrada X1=ON, todas as instruções entre a instrução MC e a MCR serão executadas.

- Quando a entrada X1=OFF, nenhuma das instruções entre a instrução MC e a MCR serão executadas; isto zera todos os dispositivos exceto os temporizadores retentivos, contadores e dispositivos acionados pelas instruções SET/RST.

- A instrução MC pode ser usada tantas vezes quanto for necessária trocando o número do dispositivo Y e M. Usar o mesmo número de dispositivo duas vezes é processado como uma bobina dupla (ver seção 1.5.2). Níveis de aninhamento podem ser duplicados, mas quando o nível de alinhamento zera, TODAS as ocorrências daquele nível zeraram e não somente aquele especificado no MC local.

Exemplo de MC por nível:

Nível N0: Linha B ativa quando X0 está ligado.

Nível N1: Linha C ativa quando X0 e X2 estão ligados.

Nível N2: Linha D ativa quando X0, X2 e X4 estão ligados.

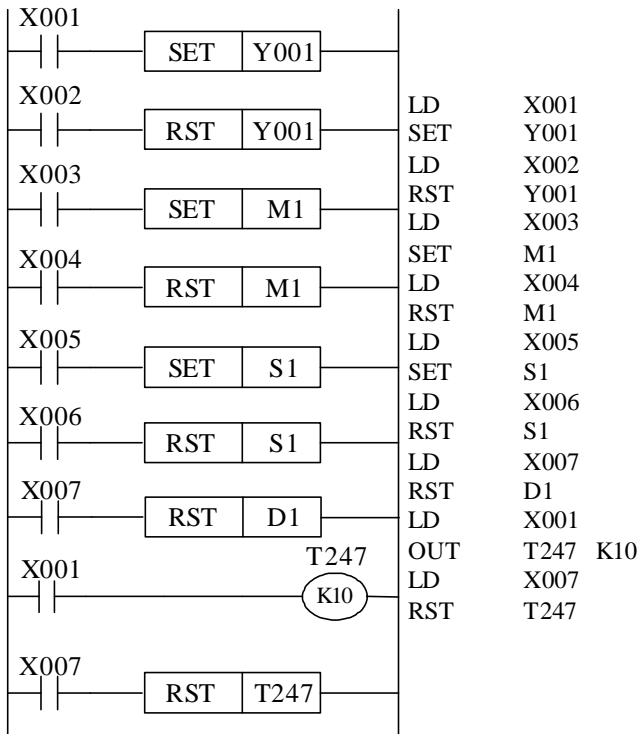
Nível N1: MCR N2 desativa o controle do nível N2. Se N0 fosse resetado, todos os níveis também seriam resetados.

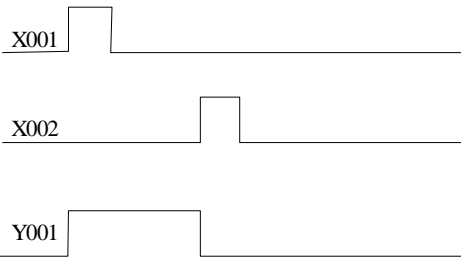
Nível N0: MCR N1 desativa o controle do nível N1.

Estado Inicial: MCR N0 desativa o controle do nível N0. A saída Y5 depende do estado da entrada X10, não importando o estado das entradas X0, X2 ou X4.

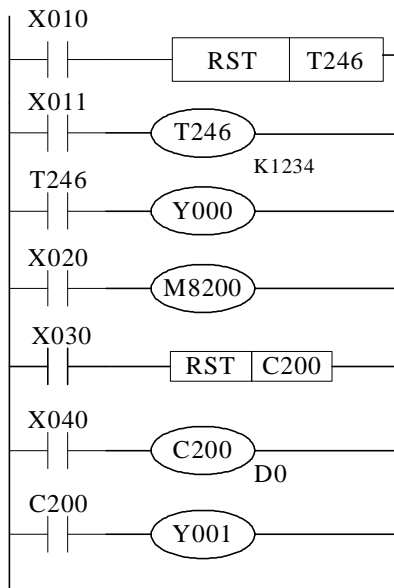
1.15 INSTRUÇÃO SET E RESET

Mnemônico	Função	Formato	Tipos de endereços	Passos do programa
[SET]	Seta um endereço de bit permanentemente em ON	X001 SET YMS	Y.M,S	Y,M:1 S, especial M : 2 D, V e Z:3
[RST]	Seta um endereço de bit permanentemente em OFF	X001 RST YMS	Y,M,S,D,V,Z	

Exemplo de programa:

Pontos básicos para lembrar:

- Colocar o X001 na posição ON faz com que o Y001 fique na posição ON. O Y001 permanece ON mesmo depois que o X001 fica na posição OFF.
- Colocar o X002 na posição ON faz com que o Y001 fique na posição OFF. O Y001 permanece na posição OFF mesmo depois que o X002 fica na posição OFF.
-As instruções SET e RST podem ser usadas para o mesmo endereço quantas vezes forem necessárias. Entretanto, a última instrução ativada determinará o estado atual do endereço
- Também é possível usar a instrução RST para zerar os dispositivos de dados de conteúdo tais como registros de dados, registro de índice etc. O efeito é similar a mover 'K0' para dentro do dispositivo de dados.

1.16 TEMPORIZADOR, CONTADOR (OUT & RESET)

Mnemônico	Função	Formato	Tipos de endereços	Passos do programa
[OUT]	Acionando saídas de temporizadores ou contadores	YMSTC	T,C	Contadores de 32 bits:5 Outros:3
[RST]	Zera o valor atual de temporização/contagem bem como as saídas dos temporizadores ou contadores	X001 RST YMS	T,C	T,C:2

1.16.1 TEMPORIZADORES BÁSICOS, TEMPORIZADORES RETENTIVOS E CONTADORES

Exemplo de programa:

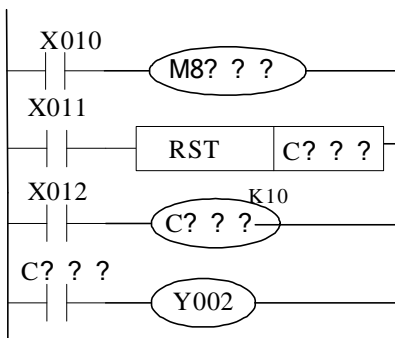
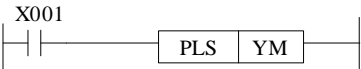
Estes dispositivos podem ser zerados a qualquer momento acionado a instrução RST (com o número do dispositivo a ser zerado). Ao zerar, todos os contatos ativos, bobinas e registros de valores atuais são zerados para o dispositivo selecionado. No exemplo, o T246, um temporizador retentivo de 1mseg, está ativo enquanto a entrada X011 está ON. Quando o valor atual do T246 alcança o valor 'K' de preset, ex: 1234, a bobina do temporizador para o T246 será ativada. Isto aciona o contato NA para a posição ON. Portanto, Y0 está na posição ON. Colocar X010 na posição ON irá zerar o temporizador T246 da forma previamente descrita. Porque os contatos T246 são zerados, a saída Y0 ficará na posição OFF.

Temporizadores retentivos:

☑ Para mais informações sobre temporizadores retentivos ver 2.9.3.

1.16.2 CONTADORES NORMAIS DE 32 BITS

O contador C200, de 32 bits, conta de forma crescente ou decrescente dependendo do estado do marcador especial M8200. No exemplo de programa (ver 1.16.1) o contador C200 está sendo usado para contar o número de pulsos na entrada X040. O contato de saída do contador C200 será ativado quanto seu valor atual atingir o preset de contagem D0. Para contadores 32 bits o preset de contagem deve ser em 32 bits, portanto D0 e D1 são ocupados. O contador será resetado quanto ativada a entrada X030.

1.16.3 CONTADORES DE ALTA VELOCIDADE

Contadores de alta velocidade têm direções de contagem que podem ser selecionadas. As direções são selecionadas acionando a bobina auxiliar especial M apropriada. O exemplo mostrado à direita funciona da seguinte maneira; quando X010 está na posição ON, acontece a contagem decrescente.

Quando X010 está na posição OFF, acontece a contagem crescente. No exemplo, quando o X11 é acionado, o valor atual de C??? é zerado e seus contatos de saída são resetados. Quando X012 está na posição ON o contador acionado está habilitado.

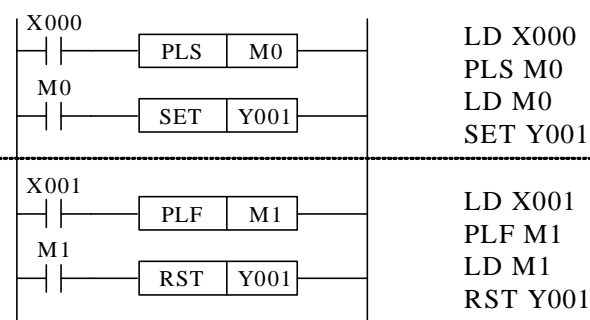
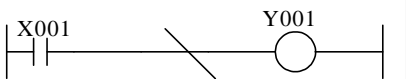
Isto significa que poderá iniciar a contar o sinal de entrada que lhe é atribuído (este não será o X012 – sinais de entrada especiais são atribuídos aos contadores de alta velocidade, ver 2.11).

Disponibilidade de endereços:

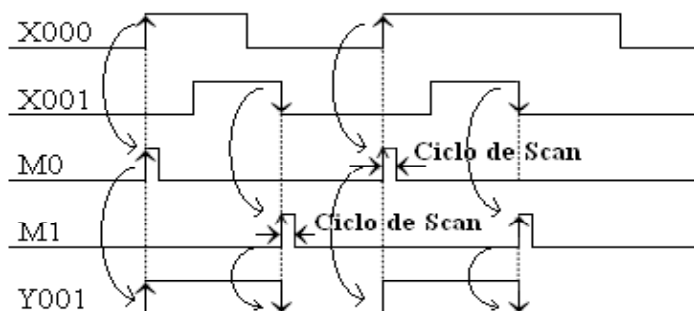
☑ Nem todos os endereços aqui identificados estão disponíveis em todos os controladores programáveis. A faixa de endereços ativos pode variar de PLC para PLC. Verificar a disponibilidade específica destes endereços no PLC selecionado antes de usar. Para mais informações sobre contadores de alta velocidade, ver 2.11. Para a faixa de endereços do PLC ver o capítulo 5.

1.17 INSTRUÇÃO PLS E PLF

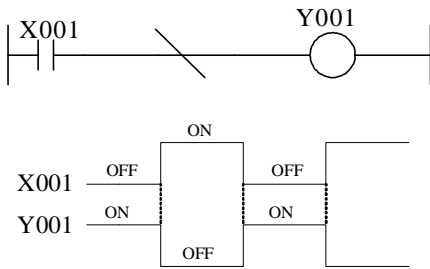
Mnemônico	Função	Formato	Tipos de endereços	Passos do programa
[PLS]	Pulso da borda de subida	X001 ─┤├─ [PLS YM]	Y,M(Marcadores especiais não são permitidos)	2
[PLF]	Pulso da borda de descida	X001 ─┤├─ [PLF YM]	Y,M(Marcadores especiais não são permitidos)	2

Exemplo de programa:

Pontos básicos para lembrar:

- Quando uma instrução PLS é executada, os endereços objetos Y e M operam por um ciclo de operação depois que o sinal de entrada estiver na posição ON.
- Quando uma instrução PLF é executada, os endereços objetos Y e M operam por um ciclo de operação depois que o sinal da entrada estiver em OFF.

Quando o estado do PLC é modificado de RUN para STOP e depois de volta para RUN com os sinais de entrada ainda ON, o PLS M0 entra em operação novamente. No entanto, se o marcador M que é usado não for M0, mas um endereço da área de memória retentiva do PLC, este será atualizado e o marcador M não seria reativado. Para endereços armazenados na área retentiva do PLC serem re-pulsados, o acionamento de entrada (ex. X0) deve estar na posição OFF durante a seqüência RUN/STOP/RUN antes que seja pulsado mais uma vez.

1.18 PORTA INVERSA (INSTRUÇÃO INV)

Mnemônico	Função	Formato	Tipos de endereços	Passos do programa
[INV]	Inverte o resultado atual das operações internas do PLC	X001 ─┤├─ / ─(Y001)	N/A	1

Exemplo de programa:

Pontos básicos para lembrar:
- A instrução INV é usada para modificar (inverter) o estado lógico da lógica *ladder* atual na posição inserida.
- Uso é o mesmo que o uso para AND e ANI visto anteriormente.

Usos para o INV

☑ Use a instrução de inversão para rapidamente mudar a lógica de um circuito complexo. Também é útil como operação inversa para as instruções de contato de pulso LDP, LDF, ANP, etc.

1.19 INSTRUÇÃO NOP (SEM FUNÇÃO)

Mnemônico	Função	Formato	Tipos de endereços	Passos do programa
[NOP]	Nenhuma operação é executada	N/A	N/A	1

Pontos básicos para lembrar:

- Escrever instruções NOP no meio de um programa minimiza o número de mudanças dos passos quando estiver mudando ou editando um programa.
- É possível mudar a operação de um circuito substituindo instruções programadas por instruções NOP.
- Trocar uma instrução LD, LDI, ANB ou uma instrução ORB por uma instrução NOP modificará consideravelmente o circuito, muito possivelmente resultando em erro.
- Depois que a função 'Clear PLC' (no TPW-03-PCLINK) é executada, todas as instruções atualmente no programa são substituídos por NOPs.

1.20 INSTRUÇÃO END (FIM DE PROGRAMA)

Mnemônico	Função	Formato	Tipos de endereços	Passos do programa
[END]	Força a varredura de programa atual a terminar	END	N/A	1

Pontos básicos para lembrar:

- Colocar a instrução END num programa força o PLC a finalizar o ciclo de scan atual e executar os processos de atualização, tanto para as entradas quanto as saídas.
- Inserir as instruções END no meio do programa ajuda no debugging/resolução de problemas no programa, pois a seção após a instrução END está desabilitada e isolada da área que está sendo verificada. Lembre-se de apagar as instruções END dos blocos que já foram verificados.
- Quando a instrução END é processada, o temporizador de watch-dog é automaticamente zerado.

Uma varredura do programa:

☑ Uma varredura de programa é a execução do programa carregado, do início ao fim. Isto inclui atualização de todas as entradas, saídas e temporizadores de watch-dog. O período de tempo para um ciclo deste ocorrer é chamado de tempo de varredura. Isto dependerá do comprimento e complexidade do programa. Imediatamente após a conclusão da varredura atual, o próximo já inicia. O processo completo é um ciclo contínuo. Atualização de entradas acontece no início de cada scan enquanto as saídas são atualizadas no final da varredura.

CAPÍTULO 2

DETALHAMENTO DOS ENDEREÇOS DE MEMÓRIA

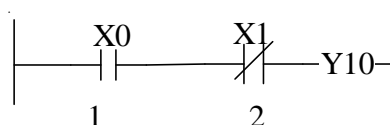
CAPÍTULO 2

DETALHAMENTO DOS ENDEREÇOS DE MEMÓRIA

2 DETALHAMENTO DOS ENDEREÇOS DE MEMÓRIA

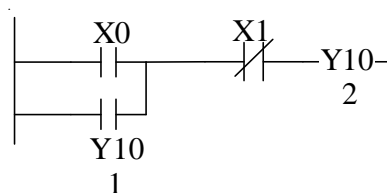
2.1 ENTRADAS

Mnemônico do endereço: X
Propósito: Representação de entradas físicas para o controlador programável (PLC)
Representação: I/P
(X) Entrada
Contato de entrada
Formas disponíveis: Contatos normalmente aberto NA (%1) e normalmente fechado NF (%2) somente.

X0 X1 Y10
1 2

Endereços numerados em: Octal, por exemplo: X0 à X7, X10 à X17.
Outros usos: Nenhum
Endereços disponíveis:
☑ Ver tabela no item 2.2. Alternativamente refira-se às tabelas relevantes para o PLC selecionado no manual de instalação.

2.2 SAÍDAS

Mnemônico do endereço: Y
Propósito: Representação das saídas físicas do controlador programável.
Representação: O/P
Out (Y)
Saída (Y)
Saída (bobina/ relé/ contato)
Formas disponíveis: Contatos normalmente aberto NA (%1), normalmente fechado NF e bobinas de saídas (%2)

X0 X1 Y10
2
Y10
1

Endereços numerados em: Octal, por exemplo: Y0 à Y7, Y10 à Y17
Outros usos: Nenhum

Endereços disponíveis:

PLC / Entradas/saídas	20 pontos	30 pontos	40 pontos	60 pontos	Máx
X (X000~X267 184 pontos)	X000~X013 12 pontos	X000~X017 16 pontos	X000~X027 24 pontos	X000~X043 36 pontos	X000~X177 128 pontos
Y(Y000~Y267 184 pontos)	Y000~Y007 8 pontos	Y000~Y005 14 pontos	Y000~Y017 16 pontos	Y000~Y027 24 pontos	Y000~Y177 128 pontos

☑ Para maiores informações sobre a disponibilidade dos endereços para cada PLC individual, ver o manual de instalação.

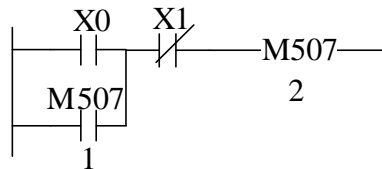
2.3 MARCADORES AUXILIARES

Mnemônico do endereço: M

Propósito: Marcador interno do status do controlador programável

Representação: Auxiliar (bobina/relé/contato/marcador) M (bobina/relé/contato/marcador) Dispositivo M (bit)

Formas disponíveis: Contatos normalmente aberto NA (%1), normalmente fechado NF e bobinas de saída (%2)

X0 X1 M507 2
M507 1

Endereços numerados em: Decimal, por exemplo: M0 a M9, M10 a M19

Outros usos: Marcadores auxiliares de estado para uso geral na programação - ver 2.3.1

Marcadores auxiliares retentivos - ver 2.3.2

Marcadores auxiliares de diagnóstico - ver 2.3.3

2.3.1 MARCADORES AUXILIARES DE ESTADO PARA USO GERAL

☑ Um certo número de marcadores auxiliares pode ser utilizado na programação do PLC. As bobinas destes marcadores são acionadas por contatos de endereços do PLC da mesma forma que os relés de saída são acionados no programa.

Todos os marcadores auxiliares têm determinado número de contatos eletrônicos NA e NF que podem ser usados na programação do PLC quando necessário. Note que estes contatos não podem acionar diretamente uma carga externa. Somente os endereços de saída física podem ser usados para fazer isto.

	Marcador auxiliar de uso geral 1	Marcadores retentivos 2	Marcadores retentivos 3	Marcadores de diagnóstico
M	M0~M499 500 pontos	M500~M1023 524 pontos	M1024~M7679 6656 pontos	M8000~M8511 512 pontos

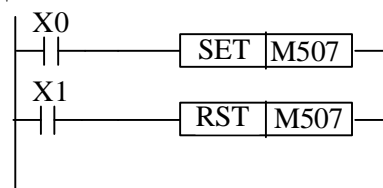
1: Não-retentivo. Porém a faixa do endereço retentivo pode ser modificada através da parametrização do PLC via o PC-LINK.

2: Retentivo. Porém a faixa do endereço retentivo pode ser modificada através da parametrização do PLC via o PC-LINK.

3: A faixa retentiva é fixa, ou seja, não pode ser modificada através do PC-LINK.

Para mais informações sobre a disponibilidade dos endereços em cada PLC individualmente ver o capítulo 5.

2.3.2 MARCADORES AUXILIARES RETENTIVOS

Existe uma determinada faixa de marcadores retentivos cujo estado é retido pela bateria de backup da memória RAM ou ainda pela memória . Se houver queda de energia todos os endereços de saída e marcadores de uso geral são desenergizados. Quando a operação é retomada o status anterior destes relés é perdido, mas o status dos marcadores retentivos é recuperado. O exemplo mostra um circuito armazenado. O relé M507 é ativado quando X0 está energizada. Se X0 é desenergizada depois do M507 ser ativado, os status ON de M507 é retido, por exemplo: O contato NA de M507 aciona a bobina M507.

Porém, o M507 é zerado(desenergizado) quando a entrada X1 é energizada, por exemplo: o contato NF deixa de conduzir.

Uma instrução SET e RST (zerar) pode ser usada para reter o status de um marcador sendo momentaneamente ativado.

Cargas externas:

☑ Marcadores auxiliares são fornecidos com inúmeros pontos de contato NA e NF. Estes estão à total disposição durante a programação do PLC. Estes contatos não podem ser usados para acionar cargas externas diretamente. Todas as cargas externas devem ser acionadas através do uso de saídas diretas (Y).

2.3.3 MARCADORES AUXILIARES DE DIAGNÓSTICO

Um PLC tem um determinado número de marcadores auxiliares especiais. Todos estes endereços possuem funções específicas e estão classificados como sendo dos dois tipos seguintes:

a) Usando contatos de marcadores auxiliares especiais

- Bobinas são acionadas automaticamente pelo PLC. Somente os contatos destas bobinas podem ser usados por um programa definido por um usuário.

Exemplos: M8000: RUN monitor (ON durante quando o PLC estiver no modo RUN)
M8002: Pulso inicial (Gera um pulso durante o primeiro ciclo de varredura do PLC)
M8012: pulso de relógio 100 mseg

b) Acionando bobinas de marcadores auxiliares especiais

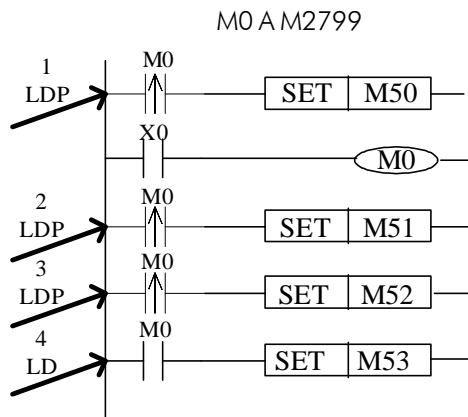
- Um PLC executa uma operação específica pré-determinada quando estas bobinas são acionadas pelo usuário.

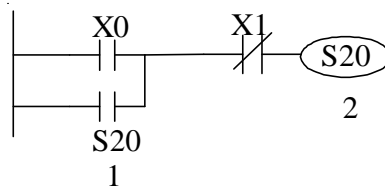
Exemplos: M8033: Todos os status de saída são retidos quando a operação do PLC é interrompida.
M8034: Todas as saídas são desabilitadas
M8039: O PLC opera sob modo de varredura constante.

Endereços disponíveis:

☑ Nem todos os PLCs compartilham a mesma faixa, quantidade ou significado operacional dos marcadores auxiliares de diagnóstico. Verificar a disponibilidade e função antes de usar algum destes endereços.

2.3.4 MARCADORES ESPECIAIS DE PULSO DE EXECUÇÃO ÚNICA

Quando usado com os contatos de pulso LDP, LDF, etc., os endereços M na faixa M2800 à M3072 têm um significado especial. Com estes endereços somente a próxima instrução após a bobina do endereço é ativada.

2.4 MARCADOR DE PASSO

Mnemônico do endereço: S
Propósito: Marcador de status de controlador programável interno
Representação: Estado (bobina/relé/contato/marcador)
S (bobina/relé/contato /marcador)
Passo STL (bobina/relé/contato /marcador)
Marcador de diagnóstico
Formas disponíveis: Contatos NA (%1) e NF e bobinas de saída (%2)

Endereços numerados em: Decimal, por exemplo: S0 a S9, S10 a S19
Outros usos: Marcador geral – marcador de estado - ver 2.4.1
Marcador de passo retentivo - ver 2.4.2
Marcadores de diagnóstico - ver 2.4.3

2.4.1 MARCADOR GERAL – MARCADOR DE ESTADO

Um determinado número de endereços pode ser usado no PLC. As bobinas destes marcadores são acionadas por contatos de endereços no PLC da mesma forma que os relés de saída são acionados no programa. Todos os marcadores de estado possuem um a quantidade de contatos eletrônicos NA e NF que podem ser usados pelo PLC quando necessário. Note que estes contatos não podem acionar uma carga externa diretamente. Somente relés de saída podem fazer isto.

Endereços disponíveis:

☑ Ver o ponto de informações 2.4.2 'Marcadores de estado' retentivos, ou ver as tabelas relevantes para o PLC selecionado no capítulo 5.

2.4.2 MARCADORES DE PASSO RETENTIVOS

Há um determinado número de endereços retentivos cujo status está retido na memória de backup ou pela EEPROM. Em caso de queda de energia todos os relés de saída e marcadores gerais serão desligados. Quando a operação for retomada o estado anterior destes relés é recuperado.

Endereços disponíveis:

Marcador do estado geral 1	Marcador de passo inicial	Para retorno zero do ITS	Marcadores de passo retentivos 2	Marcadores de diagnóstico 2
S0~S499 500 pontos	S0~S9 10 pontos	S10~S19 10 pontos	S500 ~ S4095 3596 pontos	S900 ~ S999 100 pontos

1: Não-retentivo. Porém a faixa do endereço retentivo pode ser modificada através da parametrização do PLC via o PC-LINK.

2: Retentivo. Porém a faixa do endereço retentivo pode ser modificada através da parametrização do PLC via o PC-LINK.

Para mais informações sobre a disponibilidade dos endereços em cada PLC individualmente, ver o capítulo 5.

Passos associados:

☑ Quando a instrução IST (Função de passo inicial 60) é usada os seguintes endereços de passo são automaticamente associados a operações que não podem ser mudadas diretamente por um programa de usuário:

S0 : Passo inicial de operação manual

S1 : Passo inicial de retorno zero

S2 : Passo inicial de operação automática

S10 à S19 : Alocadas para a criação da seqüência de programa de retorno zero

☑ Para mais informações sobre a instrução IST, ver 3.7.1

2.4.3 MARCADORES DE DIAGNÓSTICO

Alguns marcadores de passo podem ser usados como saídas para diagnósticos externos quando certas instruções aplicadas são usadas. Estas instruções são;

ANS função 46: Seta o diagnóstico - ver 3.5.7

ANR função 47: Reseta o diagnóstico - ver 3.5.8

Quando a função de diagnóstico é usada os marcadores de passo a serem usados estão na faixa S900 à S999 (100 pontos). Programando um circuito de diagnóstico externo como mostrado abaixo, e monitorando o registro especial de dados D8049, o bit menos significativo do marcador de diagnóstico será mostrado.

Cada um dos passos pode ser associado para significar uma condição de erro ou falha. Quando ocorre uma falha o passo associado é energizado. Se ocorrer mais que uma falha simultaneamente, a falha com o número mais baixo será mostrada. Quando a falha ativa é reconhecida, a próxima falha, com o número mais baixo será então processada.

Isto significa que para um sistema de diagnóstico corretamente priorizado as falhas mais perigosas ou que causam maiores danos deveriam ativar os marcadores de passo com endereços mais baixo. Todos os marcadores de passo usados para a função de diagnóstico caem na faixa dos registros de estado retentivos.

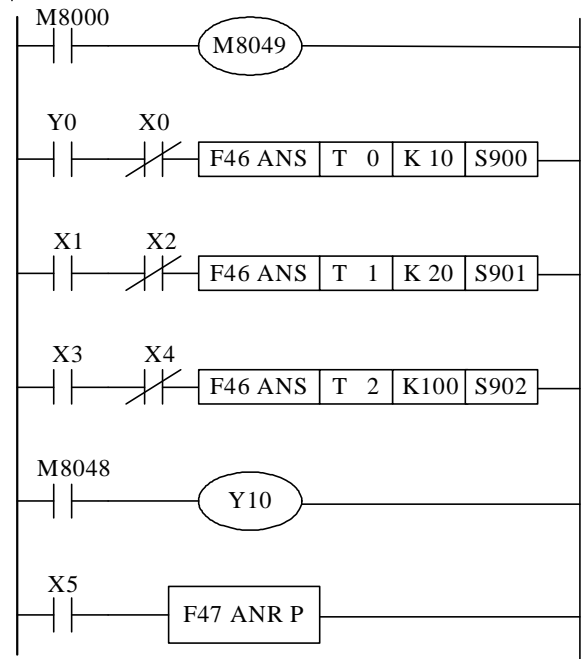
A monitoração é habilitada acionando o marcador auxiliar M8049 especial para a posição ON (energizado).

O passo S900 é ativado se a entrada X0 não é acionada no período de um segundo após a saída Y0 ir para a posição ON.

O passo S901 é ativado quando as duas entradas X1 e X2 estão desenergizadas (OFF) por mais de dois segundos. Se o tempo do ciclo da máquina controlada é menos que 10 segundos, a entrada X3 fica energizada (ON), o passo S902 será setado se X4 não for ativada dentro do tempo de ciclo desta máquina.

M8000 — M8049
Y0 X0 — F46 ANS | T 0 | K 10 | S900
X1 X2 — F46 ANS | T 1 | K 20 | S901
X3 X4 — F46 ANS | T 2 | K100 | S902
M8048 — Y10
X5 — F47 ANR P

Se qualquer passo de S900 à S999 é ativado, por exemplo: ON, o marcador auxiliar especial M8048 é ativado para ligar o indicador de falha da saída Y10.

Os passos ativados pelo programa de detecção de diagnóstico de falhas /erros dos usuários estão desenergizados (OFF) ativando a entrada X5. Cada vez que a X5 é ativada, os passos de diagnóstico ativos são zerados em ordem crescente dos números dos passos.

2.5 PONTEIROS

Mnemônico do dispositivo: P

Propósito: Controle do fluxo de programa

Representação: Ponteiro

Ponteiro do programa: P

Formas disponíveis: Etiqueta: aparece no lado esquerdo do barramento do circuito lógico quando programa é visto no modo *ladder*.

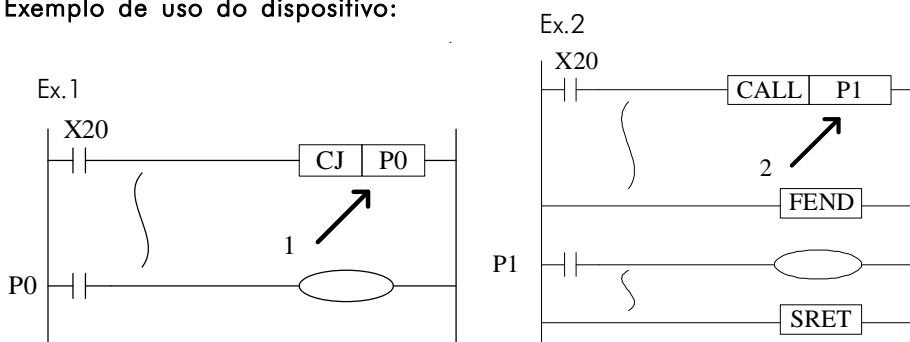
Dispositivos numerados em: Decimal, por exemplo: P0 ao P9, P10 à P19

Outros usos: Pode ser usado como definição do destino de um salto condicional (CJ função 00).

- ver 3.1.1 e Ex.:1 no dispositivo de exemplo do diagrama de uso.

Pode ser usado como identificação de uma sub-rotina

- ver 3.1.2 e Ex.:2 no dispositivo de exemplo do diagrama de uso.

Exemplo de uso do dispositivo:

Dispositivos disponíveis:

☑ O PLC tem 256 ponteiros; disponíveis na faixa P0 a P255.

Pulando para o final do programa:

☑ Ao usar instruções de salto condicionais (CJ,função 00) pode-se pular automaticamente para o final do programa usando o ponteiro P63 dentro da instrução CJ. Não é necessário rotular a instrução END com o P63.

Disponibilidade do dispositivo:

☑ Para mais informações sobre a disponibilidade do dispositivo para cada PLC individualmente, ver o capítulo 6.

2.6 PONTEIROS DE INTERRUPÇÃO

Mnemônico do dispositivo: I

Propósito: Interromper a execução do programa e direcioná-la a uma sub-rotina

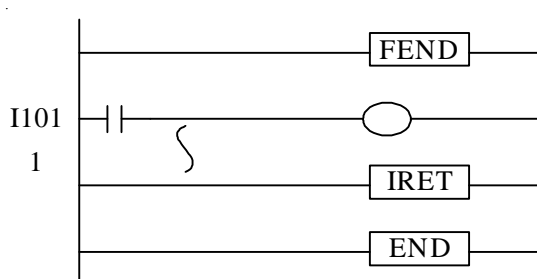
Representação: Interromper

Interrupção de alta velocidade: I

Formas disponíveis: Etiqueta: aparece no lado esquerdo do barramento do circuito lógico quando o programa é visto no modo *ladder*.

Dispositivos numerados em: Sistema especial de numeração baseado no dispositivo de interrupção usado e no método de disparo de entrada.

Outros usos: Interrupção de entrada - ver 2.6.1
Interrupção por tempo - ver 2.6.2
Interrupção de desabilitação - ver 2.6.3
Interrupção por contagem - ver 2.6.4

Exemplo de uso do dispositivo

Instruções adicionais aplicáveis:

☑ Uma interrrupção é feita de um dispositivo de interrupção, um ponteiro de interrupção e do uso de três instruções aplicadas para cada interrupção específica;

- IRET função 03: interromper o retorno - ver 3.1.4
- EI função 04: habilita a interrupção - ver 3.1.4
- DI função 05: desabilita a interrupção - ver 3.1.4

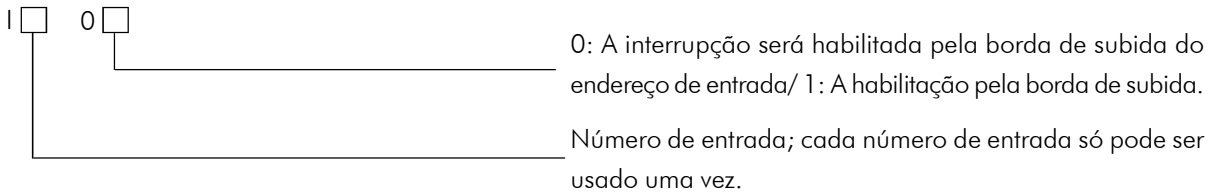
Níveis de aninhamento:

☑ Quando uma interrupção está sendo processada todas as outras interrupções são desabilitadas. Para conseguir interrupções aninhadas, a instrução EI-DI deve ser programada dentro de uma rotina de interrupção. Interrupções podem ser aninhadas em até dois níveis.

Posição do ponteiro:

☑ Ponteiros de interrupções só podem ser usados após uma instrução FEND (Função 06).

2.6.1 INTERRUPÇÕES DE ENTRADA

Identificação do número do ponteiro da interrupção:

Exemplo: I001

A seqüência programada após a etiqueta (indicada pelo ponteiro I001) é executada na borda de subida do sinal de entrada X0. A seqüência do programa retorna ao programa principal quando a instrução IRET é encontrada na rotina da interrupção.

Regras de uso:

☑ Os seguintes pontos devem ser seguidos para que uma interrupção opere;

- Ponteiros de interrupção não podem ter o mesmo número na posição ‘100’s’, por exemplo: I100 e I101 não são permitidos.

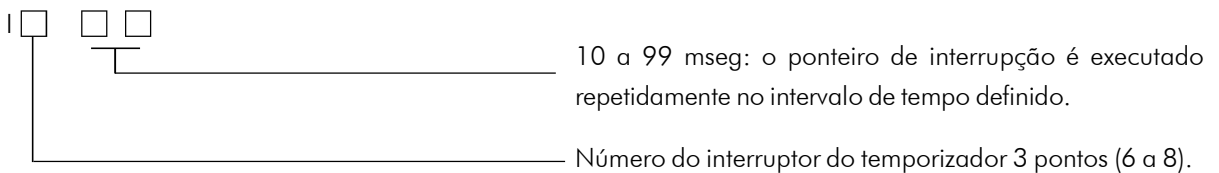
- A entrada usada para o dispositivo de interrupção não deve coincidir com as entradas já alocadas por uso pelas outras instruções de alta velocidade dentro do programa do usuário.

2.6.2 INTERRUPÇÕES DE TEMPO

Identificação do número do ponteiro da interrupção:

I☐ ☐☐

10 a 99 mseg: o ponteiro de interrupção é executado repetidamente no intervalo de tempo definido.

Número do interruptor do temporizador 3 pontos (6 a 8).

Exemplo: I610

A seqüência programada após a etiqueta (indicada pelo ponteiro I610) é executada a intervalos de 10mseg. A seqüência do programa retorna ao programa principal quando uma instrução IRET é encontrada na rotina de interrupção.

Regras de uso:

☑ Os seguintes pontos devem ser seguidos para que uma interrupção opere:

- Ponteiros de interrupção não podem ter o mesmo número na posição ‘100’s’, por exemplo I610 e I650 não são permitidos.

2.6.3 DESABILITANDO INTERRUPÇÕES INDIVIDUAIS

Dispositivos de interrupções individuais podem ser temporária ou permanentemente desabilitados acionando um marcador auxiliar especial associado. As bobinas relevantes são identificadas nas tabelas de endereços no capítulo 4. Porém, para todos os tipos de PLC o endereço principal é M805☐ desabilitará a interrupção I0☐0x.

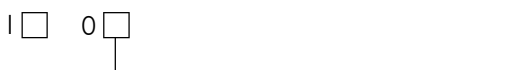
Acionando marcadores auxiliares especiais:

☑ Nunca acione uma bobina auxiliar especial sem antes verificar seu uso. Nem todos os PLCs possuem o mesmo uso para as mesmas bobinas auxiliares.

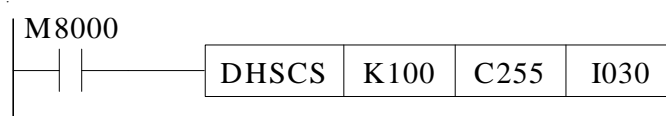
Desabilitando as interrupções do contador de alta velocidade

☑ Estas interrupções só podem ser desabilitadas como um único grupo, acionando o marcador M8059. Outros detalhes sobre interrupções de contadores podem ser encontrados nas seções seguintes.

2.6.4 INTERRUPÇÕES DE CONTAGEM

Identificação do número do ponteiro da interrupção:

Número da interrupção de contagem 6 pontos (1 a 6). Interrupções de contagem podem ser inseridas como endereços de saída para o conjunto de contadores de alta velocidade (HSCS, FNC53). Para desabilitar as interrupções de contagem, o marcador auxiliar especial M8059 deve energizado.

Exemplo:

A seqüência programada após a etiqueta (indicada pelo ponteiro I030) é executada quando o valor do contador de alta velocidade C255 alcança o valor pré-estabelecido de 100 unidades, identificado no exemplo ao lado.

Notas adicionais:

☑ Ver as seguintes páginas para mais detalhes sobre a instrução aplicada HSCS.

- Habilitação do contador de alta velocidade, HSCS FNC 53 - ver **3.6.4**

2.7 CONSTANTE K

Mnemônico do dispositivo: K

Propósito: Identificação de valores decimais constantes

Representação: Constante

K (valor/ constante)

K

Formas disponíveis: Valor de dados numéricos, quando usados para dados de 16 bits, os valores podem ser selecionados da faixa -32.768 à +32.767. Para dados de 32 bits, valores da faixa -2.147.483.648 à + 2.147.483.647 podem ser usados.

Dispositivos numerados em: N/A. Este dispositivo é um método de entrada de dados de instrução local. Não há limite quanto ao número de vezes que pode ser usado.

Outros usos: Os valores K podem ser usados com temporizadores, contadores e outras instruções.

Exemplo de uso do dispositivo: N/A

2.8 CONSTANTE H

Mnemônico do dispositivo: H

Propósito: Identificação dos valores hexadecimais constantes

Representação: Constante

H (valor/ constante)

Hex (valor/ constante)

H

Formas disponíveis: Valores de dados alpha-numéricos, por exemplo, 0 à 9 e A à F (base hexadecimal). Quando usados para dados de 16 bits, os valores podem ser selecionados da faixa 0 à FFFF. Para dados de 32 bits, valores da faixa 0 à FFFFFFFF podem ser usados.

Dispositivos numerados em: N/A. Este dispositivo é um método de entrada de dados de instrução local. Não há limite para o número de vezes que pode ser usado.

Outros usos: Valores Hex podem ser usados com instruções aplicadas.

Exemplo de uso do dispositivo: N/A

2.9 TEMPORIZADORES

Mnemônico do endereço: T

Propósito: Temporização

Representação: Temporizador(es)

T

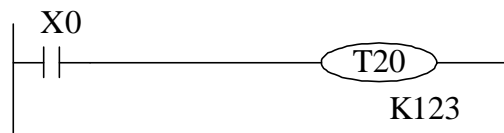
Formas disponíveis: Uma bobina acionada habilita contatos internos do PLC (contatos NA e NF estão disponíveis). Temporizadores com diversas base de tempo estão disponíveis, de 1 à 100 mseg, mas a disponibilidade e quantidade variam de PLC para PLC. As seguintes variações estão disponíveis:

Temporizadores retentivos - ver 2.9.2

Temporizadores usados em sub-rotinas de interrupção em chamadas de sub-rotinas (Função 'CALL')- ver 2.9.3

Endereços numerados em: Decimal, por exemplo: T0 à T9, T10 à T19.

Outros usos: Nenhum

Exemplo de uso de dispositivos:

Dispositivos disponíveis:

Função	Base de tempo 100ms 0.1~3276.7s	Base de tempo 10ms 0.01~327.67s	Tipo acumulativo 1ms 0.001~32.767s	Tipo acumulativo 100ms 0.1~3276.7s	Tipo 1 ms	Potenciômetro 0~1024
Geral	T0~T199	T200~T245	T246~T249	T250~T255	T256~T511	2 pontos
Para sub-rotina	T192~T199					

Precisão do temporizador:

☑ Ver 2.9.4

2.9.1 OPERAÇÃO GERAL DO TEMPORIZADOR

Temporizadores operam contando pulso de relógio (1, 10 e 100 mseg). O contato de saída do temporizador é ativado quando a contagem chega ao valor pré-definido pela constante K. A duração completa ou tempo passado para um ciclo de operação de temporizadores é calculado multiplicando o valor presente pela base de tempo do temporizador, por exemplo:

Um temporizador de base de tempo de 10 mseg com um valor de preset de 567 na verdade está operando a:

567 x 10ms

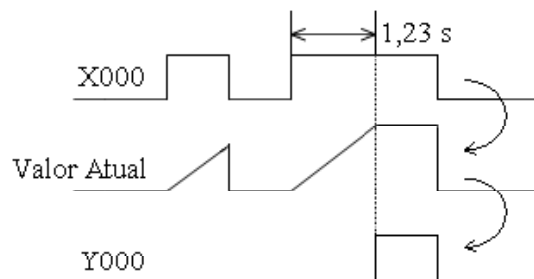
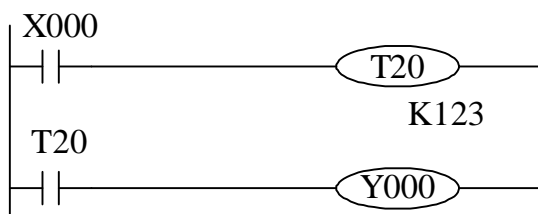
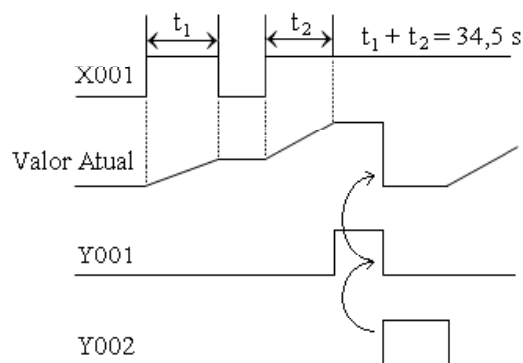
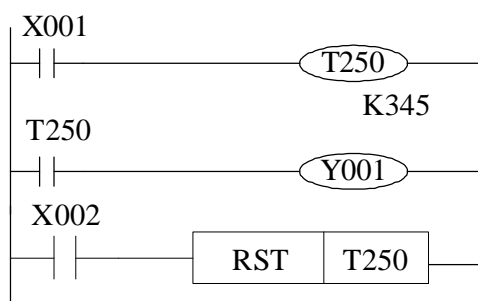
567 x 0,01s = 5,67 segundos

Temporizadores podem ser parametrizados diretamente usando a constante K para especificar a máxima duração ou indiretamente usando os dados armazenados num registro de dados (ex. D). Para a parametrização indireta, registros de dados que são retentivos são geralmente usados; isto garante que não haverá perda de dados durante situações de desenergização do PLC. Se, no entanto, a tensão da bateria usada para manter a memória de backup reduz excessivamente podem ocorrer comportamentos inesperados no temporizador.

2.9.2 TEMPORIZADORES RETENTIVOS

Um temporizador retentivo tem a habilidade de reter o valor recentemente alcançado mesmo depois que o contato de habilitação é removido. Isto significa que quando o contato de habilitação é re-estabelicido, o temporizador retentivo continuará de onde parou.

Como temporizador retentivo não é zerado quando o contato acionado é removido, ele deve ser zerado forçadamente. O diagrama a seguir mostra isto em formato gráfico.

Temporizador de operação não-retentiva

Temporizador de operação retentiva

Usando temporizadores em sub-rotinas de interrupção ou em chamadas de rotina (Função 'CALL'):

☑ Ver 2.9.4

Endereços disponíveis:

☑ Ver a tabela informativa na item 2.9.

2.9.3 TEMPORIZADORES USADOS EM SUB-ROTINAS DE INTERRUPÇÃO OU EM CHAMADA DE ROTINA (FUNÇÃO 'CALL')

Se temporizadores T192 à T199 e T246 à T249 são usados numa sub-rotina chamada pela função CALL ou numa rotina de interrupção, a ação de temporização é atualizada no ponto em que a instrução END é executada. O contato de saída é ativado quando uma instrução de bobina ou uma instrução END é processada uma vez que o valor atual dos temporizadores alcança o valor pré-programado(máxima duração).

Temporizadores além dos especificados acima não podem funcionar corretamente dentro das circunstâncias especificadas.

Quando um temporizador de interrupção (base de tempo de 1 mseg) é usado numa rotina de interrupção ou numa sub-rotina chamada pela função 'CALL', o contato de saída é ativado quando a primeira instrução de bobina daquele temporizador é executado depois que o temporizador chegou no seu valor pré-programado (final da temporização).

2.9.4 PRECISÃO DO TEMPORIZADOR

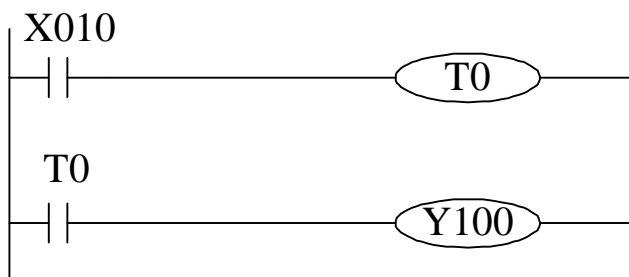
A precisão do temporizador pode ser afetada pela configuração do programa. Isto é, se um contato de temporizador é usado antes da sua bobina associada, então a precisão do temporizador é reduzida.

As formulas a seguir informam erros máximos e mínimos para determinadas situações.

Entretanto, um erro médio esperado seria aproximadamente;

1.5 x O ciclo de varredura do programa

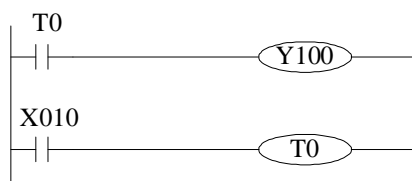
Condição 1:

O contato do temporizador aparece depois da bobina do temporizador.

Erro máximo de temporização:

2 x o ciclo de varredura + o tempo do filtro de entrada

Erro mínimo de temporização:

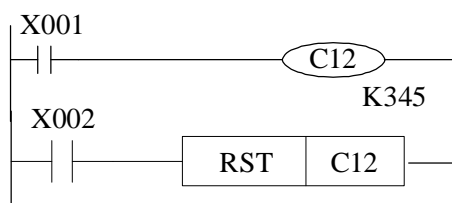
Tempo do filtro de entrada – a resolução do temporizador

Condição 2:
O contato do temporizador aparece antes da bobina do temporizador.

Erro máximo de temporização:
3 x ciclo de varredura + tempo do filtro de entrada
Erro mínimo de temporização:
Tempo do filtro de entrada - a resolução do temporizador
Precisão do temporizador interno:

A precisão atual dos elementos de temporização no hardware do PLC é ± 10 pulsos por milhão de pulsos. Isto significa que se um temporizador de 100 mseg é usado para temporizar um único dia, no final do dia o temporizador estará dentro 0,8 segundos das verdadeiras 24 horas ou 86.400 segundos. O temporizador será processado aproximadamente 864.000 vezes.

2.10 CONTADORES

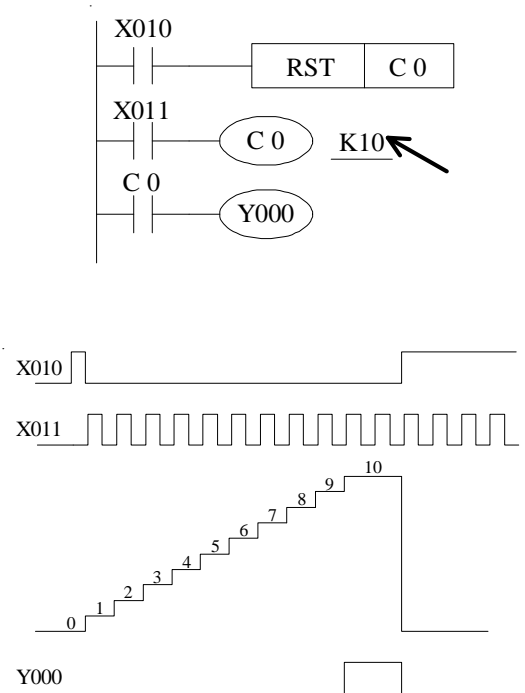
Mnemônico do endereço: C
Propósito: Contagem de eventos
Representação: Contador(es)
C
Formas disponíveis: Uma bobina acionada habilita contatos internos do PLC (contatos NA e NF disponíveis). Várias resoluções de contador são possíveis, inclusive;
Contadores de uso geral/retentivos até 16 bits - ver 2.10.1
Contadores bidirecionais de uso geral/retentivos de 32 Bits – ver 2.10.2
(A disponibilidade e uso de todos estes contadores é específico do PLC – verificar disponibilidade antes de usar)
Endereços numerados em: Decimal, por exemplo: C0 à C9, C10 à C19
Outros usos: Nenhum

Exemplo de uso do dispositivo:

Endereços disponíveis:

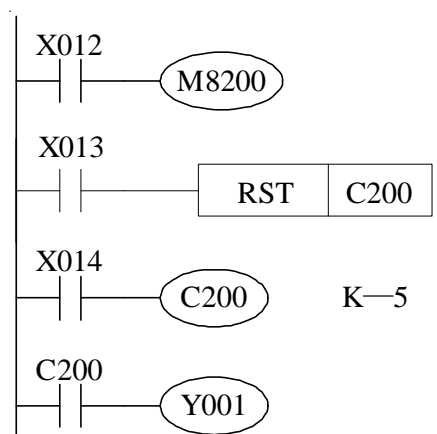
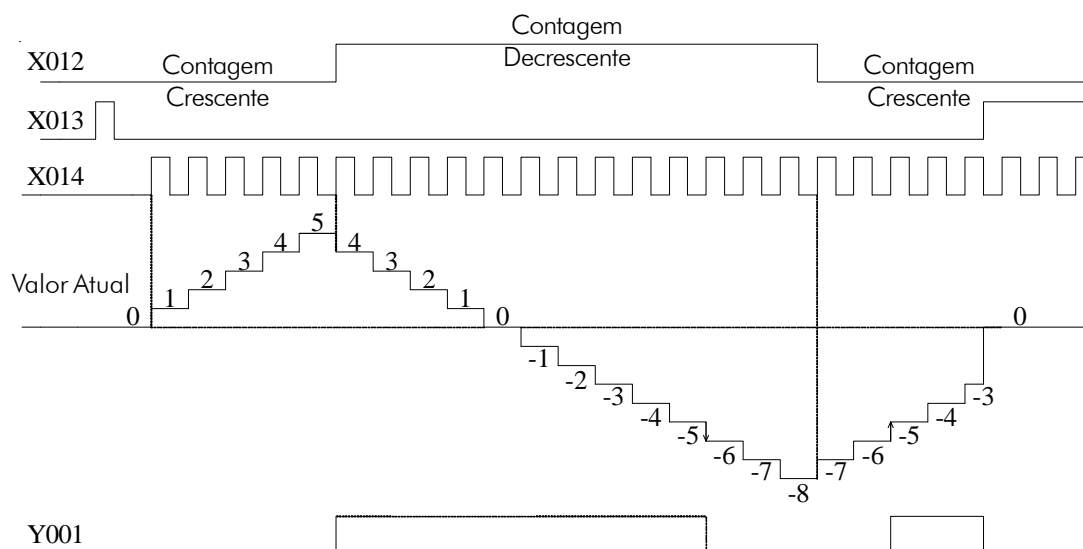
Contador de uso geral 16 bits unidirecional 0 ~ 32.767	Contador retentivo 16 bits unidirecional 0 ~ 32.767	Contador retentivo 32 bits -2.147.483.648 ~ +2.147.483.647
C0~C099	C100~C199	C200-C255

Contadores de alta velocidade:
☑ Para contadores de alta velocidade ver 2.11
Definição das faixas para contadores:
☑ Contadores de 16 bits: -32.768 à +32.767
☑ Contadores de 32 bits: -2.147.483.648 à +2.147.483.647

2.10.1 CONTADORES USO GERAL/ RETENTIVOS 16 BITS UNIDIRECIONAL

O valor atual do contador aumenta cada vez que a bobina C0 é energizada pela entrada X011. O contato de saída é ativado quando a bobina é energizada pela décima vez (ver diagrama). Depois deste, os dados do contador permanecem inalterados mesmo que X011 seja energizado. O valor atual do contador é zerado quando a instrução RST é executada habilitando a entrada X010 como no exemplo. O contato de saída Y000 também é zerado ao mesmo tempo. Contadores podem ser programados diretamente usando a constante K ou indiretamente usando dados guardados num registro de dados (ex. D). Numa programação indireta, a designação do D10, por exemplo, que contém o valor “123” tem o mesmo efeito que uma programação de “K123”. Se um valor maior que a programação do contador é escrito num registro de valor atual, o contador conta para cima quando a próxima entrada está na posição ON. Isto é o mesmo para todos os tipos de contadores. Geralmente a freqüência de entrada do contador deveria ser aproximadamente vários ciclos por segundo.

Contadores retentivos:
☑ Contadores retentivos podem reter seu status de informação até mesmo depois que o PLC for desenergizado. Isto significa que ao ser energizado novamente, os contadores retentivos podem imediatamente continuar a contagem de onde eles estavam no momento em que o PLC foi desligado.
Endereços disponíveis:
☑ Ver as informações da tabela na página 40.

2.10.2 CONTADORES BIDIRECIONAIS DE USO GERAL/ RETENTIVOS 32 BITS

O contador mostrado no exemplo abaixo ativa quando sua bobina é acionada, por exemplo, a bobina C200. Em todas as ocasiões, a contagem ocorre quando há uma transição positiva na entrada X014.

O endereço de saída C200 energizado quando o valor atual aumenta de "-6" para "-5". Entretanto, se o valor de contagem diminuir de "-5" para "-6" a bobina do contador será zerada. O valor atual dos contadores aumenta ou diminui independente do estado do contato de saída (ON/OFF). No entanto, se o contador contar um valor além de +2.147.483.647, o valor atual mudará automaticamente para -2.147.483.648. Do mesmo modo, se a contagem for decrescente, ao atingir -2.147.483.648, o contador mudará o valor atual para +2.147.483.647. Este tipo de técnica de contagem é típica em "contadores cíclicos". O valor atual do contador ativo pode ser zerado resetando a bobina do contador; no programa de exemplo, energizando a entrada X013, ela aciona a instrução RST. A direção da contagem é designada pelos marcadores auxiliares especiais M8200 à M8255.

Contadores retentivos:

☑ Contadores retentivos podem reter seu status de informação até mesmo depois que o PLC for desligado. Isto significa que ao ser ligado novamente, os contadores retentivos podem imediatamente continuar a contagem de onde eles estavam no momento em que o PLC foi desenergizado.

Endereços disponíveis:

☑ Ver a tabela de informações na página 40.

Selecionando a direção da contagem:

☑ Se o respectivo endereço M82 □□ do contador C2 □□ for energizado, o contador executará contagem decrescente. O inverso é verdadeiro, o contador contará crescentemente quando seu respectivo endereço M82 □□ estiver desenergizado.

2.11 CONTADORES DE ALTA VELOCIDADE

Mnemônico do endereço: C

Propósito: Contagem de sinais de alta freqüência

Representação: Contador (es)

C

Contador(es) de alta velocidade

Contadores de fase

Formas disponíveis: Uma bobina acionada habilita contatos internos de PLC (contatos NA e NF disponíveis).

Há vários tipos de contadores de alta velocidade disponíveis mas a quantidade e função variam de PLC para PLC.

Endereços numerados em: Decimal, por exemplo: C235 à C255

Outros usos: Nenhum

Exemplo de uso do dispositivo: Para exemplos de cada uma das normas disponíveis ver as seções relevantes.

2.11.1 OPERAÇÃO BÁSICA DO CONTADOR DE ALTA VELOCIDADE

Apesar dos contadores C235 à C255 (21 pontos) serem todos contadores de alta velocidade, todos eles usam as mesmas entradas de alta velocidade. Portanto, se uma entrada já está sendo usada por um contador de alta velocidade, a mesma não pode ser usada por nenhum outro contador de alta velocidade ou para nenhum outro propósito, por exemplo: como uma entrada de interrupção.

A seleção dos contadores de alta velocidade não é livre, ela depende diretamente do tipo de contador requerido e quais entradas estão disponíveis.

Tipos de contadores disponíveis;

a) 1 canal: C235 à C245

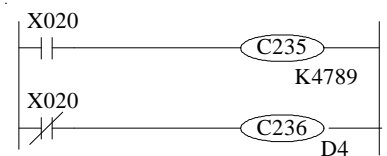
b) 1 canal bidirecional: C246 à C249

c) 2 canais bidirecional: C251 à C254

Note que TODOS estes contadores são endereços de 32 bits.

Contadores de alta velocidade operam pelo princípio de interrupções. Isto significa que eles são eventos "trigados" e são executados independentes do tempo de ciclo. A bobina do contador selecionado deveria ser acionada continuamente para indicar que esse contador e suas entradas associadas estão reservadas e que outros processos de alta velocidade não devem coincidir com eles.

Exemplo:

Quando a entrada X020 é energizada, o contador de alta velocidade C235 é habilitado. O contador C235 corresponde à entrada de contagem X000. X020 NÃO é o sinal contado. Este é apenas a habilitação contínua mencionada anteriormente. O X000 não precisa ser incluído no programa.

A associação da entrada relacionada ao contador é feita pelo hardware interno do PLC e não pode ser modificada pelo usuário.

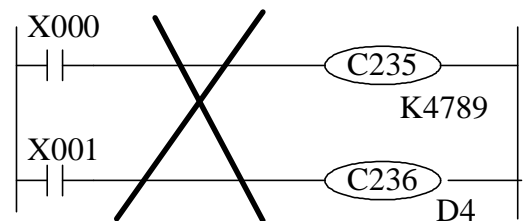
Quando X020 é desenergizado, a bobina C235 também será e a bobina C236 irá ligar. O contador C236 é associado à entrada X001, novamente a entrada X020 NÃO é a entrada de contagem.

A associação dos contadores e endereços de entrada depende do PLC selecionado. Isto é explicado mais tarde nas seções relevantes.

Acionando bobinas de contadores de alta velocidade:

☑ As entradas de contagem NÃO são usadas para acionar as bobinas dos contadores de alta velocidade.

Isto porque as bobinas do contador precisam estar continuamente habilitadas para executar a contagem das entradas de alta velocidade associadas.

Portanto, um contato normal, que não o de contagem rápida, deveria ser usado para habilitar o contador rápido. O ideal é que o marcador auxiliar especial M8000 fosse usado para tal função, entretanto, isto não é obrigatório.

	1 canal / 1 entrada de contagem											1 canal / 2 entradas de contagem				2 canais / 2 entradas de contagem			
	C235	C236	C237	C238	C239	C240	C241	C242	C243	C244	C245	C246	C247	C248	C249	C251	C252	C253	C254
X000	U/D						U/D			U/D		U	U		U	A	A		A
X001		U/D					R			R		D	D		D	B	B		B
X002			U/D					U/D			U/D		R		R		R	A	R
X003				U/D				R			R			U				B	
X004					U/D				U/D	S				D				R	
X005						U/D			R		S			R	S				S

U: Contagem para cima — R: Reset (Zeramento) — A: Fase A
D: Contagem para baixo — S: Set (Habilitação) — B: Fase B

2.12 REGISTRADOR DE DADOS

Mnemônico do endereço: D
Propósito: Um endereço de "word" capaz de armazenar dados numéricos ou padrões de 16 / 32 bits
Representação: Dados (registro/ endereço/ "word")
D (registro)
D
Word
Formas disponíveis: Registros de uso geral
Registros retentivos
Registros de diagnóstico especial
Registros de arquivos
Endereços numerados em: Decimal, por exemplo: D0 à D9, D10 à D19
Outros usos: Pode ser usado na programação indireta de contadores e temporizadores
Exemplo de uso do dispositivo: Nenhum

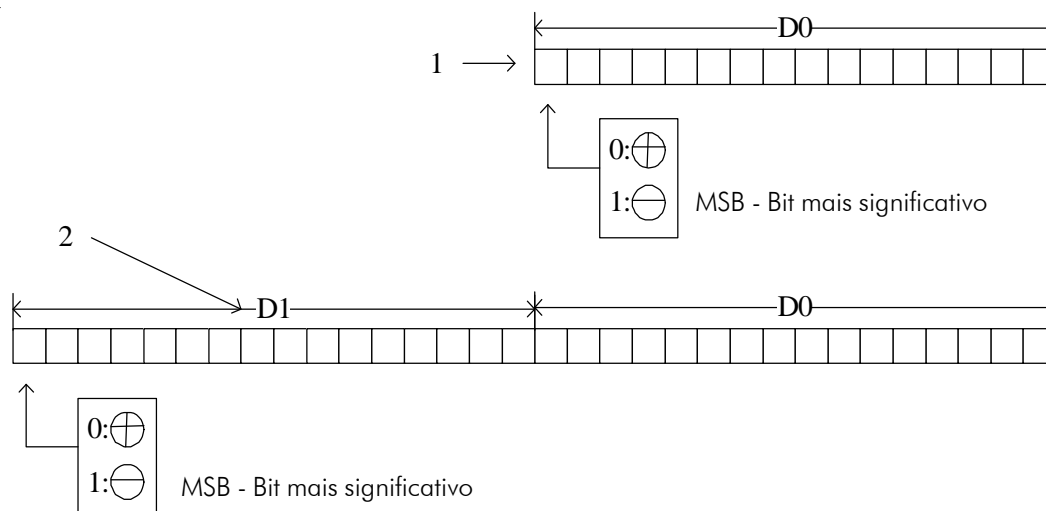
Endereços disponíveis:

Registradores de uso geral - configuráveis como retentivos/não-retentivos	Registradores de uso geral - retentivos	Registradores de diagnóstico especial
D0 ~ D511 512 pontos	D512 ~ D7999 7488 pontos	D8000 ~ D8511 512 pontos

2.12.1 REGISTROS DE USO GERAL

Registros de dados, como o próprio nome sugere, armazenam dados. Os dados armazenados podem ser interpretados como um valor numérico ou como uma série de bits, informando status ON ou OFF de cada um deles.

Um único registro de dados contém 16 bits ou uma "word". No entanto, dois registros de dados consecutivos podem ser usados para formar um endereço de 32 bits mais comumente conhecido como uma "double word".

Se o registrador de dados está sendo utilizado para armazenar um valor numérico, então o bit mais significativo (MSB) é usado para indicar se o numeral armazenado é positivo ou negativo. Como endereço de bits só pode informar status do tipo ON ou OFF.

O diagrama acima mostra as configurações de registrador tipo "word – 16 bits" e "double word – 32 bits". No desenho acima repare que o bit 15 (último bit a esquerda da word) do registrador D0 não é mais o 'Bit mais significativo', isto porque ele é agora considerado como parte de uma "double word" de 32 bits. O MSB será sempre encontrado nos 16 bits mais altos, por exemplo: neste caso o D1. Ao especificar um registro de dados de 32 bits numa instrução de programa, o endereço mais baixo é sempre usado para declarar a "double word", por exemplo: se o exemplo acima fosse escrito como um operando instrucional de 32 bits ele seria identificado como D0. O segundo registro, D1, seria automaticamente associado.

Uma vez que os dados são escritos num registro de dados geral, eles se mantêm inalterados até que sejam reescritos. Quando o PLC sai da posição RUN para STOP, todos os registros dados gerais têm seus conteúdos atuais reescritos com um 0 (zero).

Retenção de dados:

☑ Dados podem ser retidos nos registros de uso geral quando o PLC alterar o modo de operação de RUN para STOP se o marcador auxiliar especial M8033 está habilitado.

Atualizações dos registradores de dados:

☑ Escrever um novo valor de dados num registrador resultará na atualização do registro com o novo valor de dados no final da varredura atual do programa.

2.12.2 REGISTROS DE DIAGNÓSTICO ESPECIAL

Registradores especiais são usados para controlar ou monitorar vários modos ou endereços dentro do PLC. Dados escritos nestes registros são programados nos valores default e carregados quando o PLC é energizado.

- Nota: Quando o PLC é alimentado, todos os registradores são primeiramente zerados e depois os valores default são automaticamente carregados aos registros endereçados pelo software do sistema. Por exemplo, os dados do temporizador de "watch-dog" é escrito para D8000 pelo sistema operacional do PLC. Para mudar a programação, o usuário deve escrever o valor solicitado sobre o valor que está atualmente armazenado no D8000.

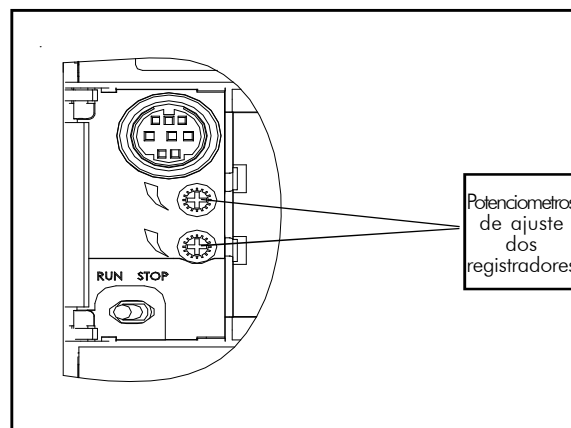
Dados armazenados nos registros de diagnóstico especial permanecerão inalterados mesmo quando o PLC sai do modo de operação para outro.

Uso dos registros de diagnóstico:

Endereços de diagnóstico não devem, sob hipótese alguma, ser usados para um propósito diferente do explanado neste manual. Ver o capítulo 4 para tabelas contendo dados e descrições de todos os endereços disponíveis para cada PLC.

2.12.3 REGISTRADORES EXTERNAMENTE AJUSTADOS

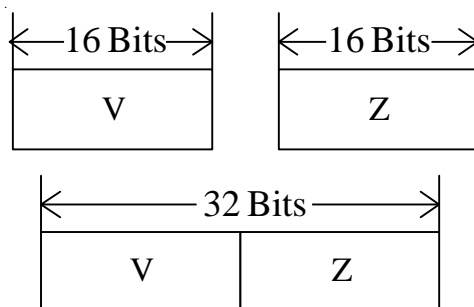
O PLC tem "trim-pots" externos que são usados para ajustar o conteúdo de certos registradores dedicados. O conteúdo destes registros pode variar de 0 a 1023. Esta é uma característica interna e não requer nenhuma programação adicional.

Existe também uma unidade adicional que fornece a mesma função. Para usar esta unidade, as instruções aplicadas VRRD função 85 (Volume Read) e VRSC função 86 (Volume Scale) devem usadas.

Número de potenciômetros	2 pontos: Existentes na unidade básica 6 pontos: Disponíveis ao utilizar módulo de expansão
Registradores utilizados	Selecionados pelo usuário quando as instruções aplicadas VRRD e VRSC são usadas

Utilização:

☑ Estes potenciometros são freqüentemente usados para variar a programação dos temporizadores, mas podem ser usados em qualquer aplicação onde os registradores de dados são normalmente encontrados, por exemplo: programando contadores, fornecendo dados básicos, até a seleção de operações poderia ser executada usando esta opção.

2.13 REGISTRADORES DE ÍNDICE

Mnemônico do endereço: V,Z
Propósito: Modificar um endereço específico declarando um "offset".
Representação: (V/Z) Registro
Índice (registro/endereçamento/modificador)
Offset(s) (registro/endereçamento/modificador)
Índices
Modificador
Formas disponíveis:
Para dados de 16 bits V ou Z
(2 tipos de endereços)
Para dados de 32 bits V e Z combinados
(1 tipo de endereço - Z é especificado)
A operação é similar para todos os registros de dados.

Endereços numerados em: N/A. Há 32 dispositivos V0 - V15 e Z0 - Z15
Outros usos: Pode ser usado para modificar os seguintes endereços em determinadas condições;
X, Y, M, S, P, T, C, D, K, H, KnX, KnY, KnM, KnS
Exemplo de uso do dispositivo:
O programa mostrado a seguir transfere dados do D5V para o D10Z.
Se os dados contidos no registro V são iguais a 8 e os dados no registro Z são iguais a14, então:
Exemplo de uso do dispositivo:
O programa mostrado a seguir transfere dados do D5V para o D10Z.
Se os dados contidos no registro V são iguais a 8 e os dados no registro Z são iguais a14, então:
V = 8
D5V
D5 +8 =13 → D13
Z = 14
D10Z
D10 + 14 = 24 → D24

Portanto, os endereços usados depois dos registradores V e Z levados em consideração são: D13 e D24 e não D5 e D10 respectivamente.

Uso de registradores com parâmetros de instruções aplicadas:

☑ Todos os parâmetros de instruções aplicadas deveriam ser vistos como podendo usar registros de índice para modificar o operando exceto onde especificado ao contrário.

2.13.1 MODIFICANDO UMA CONSTANTE

Constantes podem ser modificadas tão facilmente quanto registros de dados ou endereços de bits. Se, por exemplo, a constante K20 for escrita K20V o resultado final seria igual a:

K20 + o conteúdo de V

Exemplo:

Se V=3276 então K20V → K 20 + V 3276 = 3296

2.13.2 USO ERRADO DOS REGISTRADORES

Modificar endereços Kn quando o Kn forma parte da descrição de um endereço como KnY, não é possível. Por exemplo, o uso dos seguintes registradores é permitido;

K3Z

K1M10V

Y20Z

Declarações da forma:

K4ZY30

não são aceitáveis.

☑ Registradores não podem ser usados para parâmetros inseridos em quaisquer das 20 instruções básicas, por exemplo: LD, AND, OR etc.

2.13.3 USANDO REGISTROS DE ÍNDICE MÚLTIPLO

O uso de registros de índice múltiplo é por vezes necessário em programas maiores ou programas que lidam com uma grande quantidade de dados. Não há problema nenhum do ponto de vista do PLC em usar os registros V e Z várias vezes no decorrer de um programa. O que precisa ser notado é que por vezes o usuário ou a pessoa de manutenção, ao fazer a leitura destes programas pode se confundir, pois nem sempre está claro qual o valor atual de V ou Z.

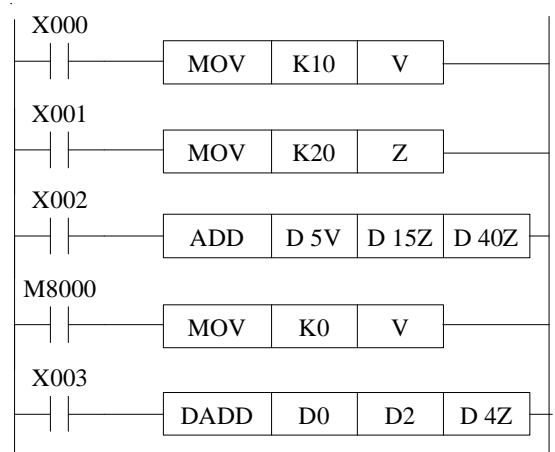
Exemplo:

V = 10 (K10)

Z = 20 (K20)

D5V = D15 (D5 + V = D5 + 10 = D15)

D15Z = D35 (D15 + Z = D15 + 20 = D35)

D40Z = D60 (D40 + Z = D40 + 20 = D60)

Os registros V e Z são inicialmente programados para K10 e K20 respectivamente.
O conteúdo de D15 é adicionado ao do D35 e armazenado no D60.
V é, então, zerado e tanto V como Z são usados na adição de "double word" (DADD).
O conteúdo de D1, D0 é, então, adicionado ao D3, D2 e finalmente armazenado no D25, D24.

2.14 BITS, WORDS, BCD E HEXADECIMAL

A seção a seguir detalha os tópicos gerais com relação a uma boa compreensão dos endereços. Esta seção está dividida em várias partes menores, cada uma cobrindo um tópico ou pequeno grupo de tópicos.

Endereços disponíveis:

☑ Para endereços específicos disponíveis para PLC, ver o capítulo 5.

2.14.1 ENDEREÇOS DE BITS, INDIVIDUAIS OU AGRUPADOS

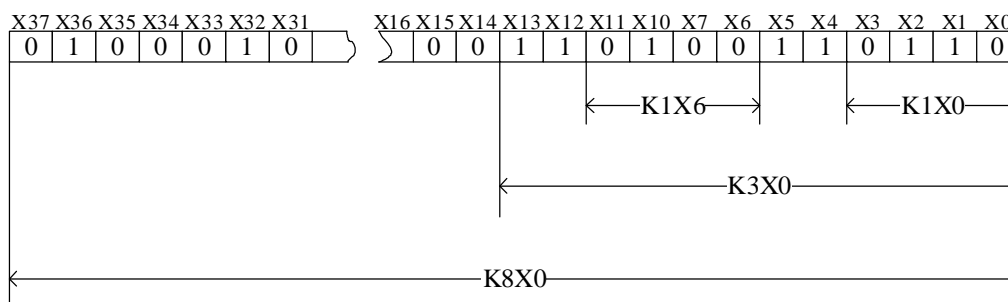
Endereços como X, Y, M e S são memórias de bits. Endereços de bits são biestáveis, o que significa que podem armazenar somente dois estados, ON e OFF ou 1 e 0. Estas memórias podem ser agrupados para formar representações de dados maiores. Por exemplo, endereços de 8 bits consecutivos são às vezes conhecidos como um byte. Mais ainda, endereços de 16 bits consecutivos são conhecidos como uma "word" e endereços de 32 bits consecutivos são uma "double word".

O PLC identifica grupos de endereços de bits que devem ser analisados bit-à-bit, permitindo que um único bit dentro do registrador possa ser utilizado na programação. Este bit seria identificado da forma KnP onde "P" representa o número do primeiro bit dentro do registrador a ser analisado. A porção Kn do endereçamento identifica a quantidade de grupos de bits que serão analisados, onde "n" pode ser um número de 0 à 8. Cada dígito "n" representa grupos de 4 bits em 4 bits. Por exemplo, K1 = 4 bits e K8 = 32 bits. Portanto, todos os grupos de bits são divisíveis por 4.

Associando bits agrupados:

Como já foi explicado, endereços de bits podem ser agrupados em grupos de 4 bits em 4 bits. O "n" em KnM0 define o número de grupos de 4 bits a ser combinado para operação de dados. K1 e K4 são permitidos para operações de dados de 16 bits, mas K1 à K8 são válidos para operações de 32 bits.

O K2M0, por exemplo, identifica 2 grupos de 4 bits iniciando do bit "0" do registrador; M0 à M3 e M4 à M7, dando um total de 8 bits ou 1 byte.

O diagrama abaixo identifica mais exemplos do uso de Kn□ .

K1X0 : X0 à X3 → 4 bits a serem analisados começando pelo endereço X0
K1X6 : X6 à X11 → 4 bits a serem analisados começando pelo endereço X6
K3X0 : X0 à X13 → 12 bits a serem analisados começando pelo endereço X0
K8X0 : X0 à X37 →32 bits a serem analisados começando pelo endereço X0

Transferindo endereços de bits agrupados:

☑ Transferir dados envolve levar dados de uma fonte e um determinado destino, considerando apenas a porção definida na instrução de transferência, os demais bits serão ignorados. Por exemplo: se o K3M20 é movido para K1M0 então somente os grupos de bits M20 à M23 ou K1M20 serão na verdade movidos. Os dados K2M24 ou M24 à M31 restantes são ignorados.

Associando I/O:

☑ Qualquer valor tirado da faixa de endereços disponíveis pode ser usado como o 'marcador' do endereço principal de um grupo de bits. Entretanto, recomenda-se sempre iniciar com 0 (zero) no lugar de dígito mais baixo dos endereços X e Y (X0, X10, X20, etc.). Para os endereços M e S, o uso de um múltiplo de "8" é o mais eficiente. Porém, como o uso de tais números pode levar a confusão na associação dos números dos endereçamentos, recomenda-se usar um múltiplo de "10". Isto permitirá uma boa correlação para os endereços X e Y.

2.14.2 ENDEREÇOS DE WORD

Endereços de Word como T, C, D, V e Z podem armazenar dados sobre um evento em particular ou uma ação no PLC. Na maioria das vezes estes endereços são registradores de 16 bits. Porém, certas variações possuem capacidades de 32 bits, assim como podem trabalhar como pares de registros de dados consecutivos ou registros V e Z combinados.

Pode parecer estranho citar o tamanho de um endereço "word" em bits, mas não é tão estranho quando se considera que o bit é a menor unidade de dados dentro do PLC. Portanto, ao identificar tudo em formato de bit uma denominação comum está sendo usada, tornando a comparação muito mais fácil.

Conseqüências adicionais desta interpretação de bits é que os dados, em si, podem ser interpretados de forma diferente. O padrão físico dos bits ativos pode ser a característica importante ou talvez a interpretação numérica do padrão dos bits pode ser a chave do programa. Tudo se resume em como a informação é lida.

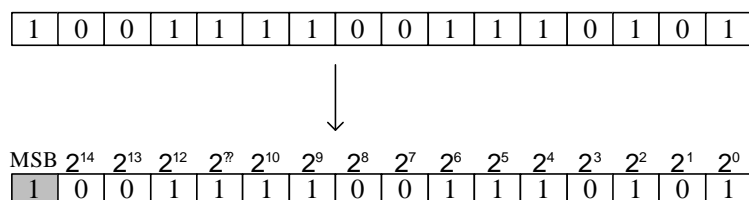
2.14.3 INTERPRETANDO DADOS DE WORD

Como os dados de word podem ser lidos de várias maneiras a significância de certas partes dos dados da word pode mudar. Os PLCs podem ler os dados de word como:

- Um simples conjunto de bits
- Um número decimal
- Um número hexadecimal
- Ou como um número BCD (Binary Coded Decimal) Decimal codificado para binário

Os exemplos a seguir mostrarão como um mesmo conjunto de dados pode se tornar algo diferente dependendo totalmente da forma como a informação é lida ou interpretada.

a) Considerando um simples conjunto de bits

O padrão de bits a seguir não significa nada – é simplesmente um registrador que contém 16 endereços de bits e podem armazenar dois estados. Alguns dos endereços são aleatoriamente programados para um dos estados. Entretanto, se os bits forem analisados com o valor de seu peso na base 2, o dado armazenado no registrador de 16 bits pode ser utilizado como um número decimal, basta considerar apenas os bits com estado lógico em 1, depois converter o seu valor em decimal considerando seu peso na base 2 e somar o valor de todos os bits. Por exemplo:

Valor decimal = $(2^0 \times 1) + (2^2 \times 1) + (2^4 \times 1) + (2^5 \times 1) + (2^6 \times 1) + (2^9 \times 1) + (2^{10} \times 1) + (2^{11} \times 1) + (2^{12} \times 1)$

Valor decimal = 7797

No entanto, este valor está incorreto!

Há um endereço de bit que foi destacado (bit 15). Se sua identificação for cuidadosamente estudada, verificar-se-á que diz MSB. Este é o bit mais significativo. Este bit sozinho determinará se o dado calculado é um número positivo ou negativo. Neste exemplo, o MSB é igual a 1. Isto significa que o número calculado é negativo.

A resposta, entretanto, não é -7797.

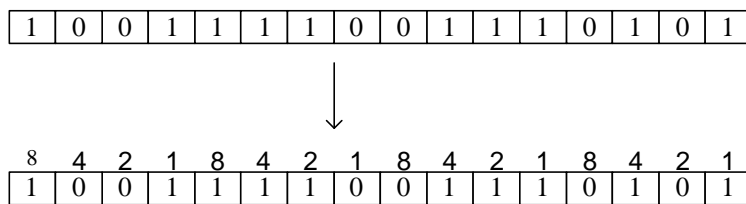
A razão pela qual não é -7797 é porque um valor negativo é calculado usando complemento de "2" (descrito depois) mas pode ser rapidamente calculado da seguinte maneira:

Por ser um número negativo, a base é programada em -32768. Este é o menor número disponível com dados de 16 bits. A este valor é adicionado a soma positiva dos bits ativados, por exemplo: -32768 +7797.

A resposta correta é, portanto, -24971.

Lembre-se que esta é agora uma representação decimal do padrão original de um registrador de 16 bits. Se o padrão original fosse acessado novamente como um número hexadecimal a resposta seria diferente.

a) Uma visão hexadecimal

Tomando o mesmo padrão de bits original usado no ponto "A" e agora adicionando uma notação hexadecimal ao invés da notação na base binária (base 2) o significado dos bits armazenados no registrador torna-se:

Valor hexadecimal = (1 x 8) + (1 x 1)) , ((1 x 8) + (1 x 4) + (1 x 2) , (1 x 4) + (1 x 2) + (1 x 1) , (1 x 4) + (1 x 1)

Valor hexadecimal = 9E75

Duas coisas tornam-se óbvias imediatamente após uma conversão hexadecimal. A primeira é que o valor dos bits como número hexadecimal é sempre positivo.

A segunda é que aparece um "E" nos dados calculados. Isto é na verdade aceitável já que hexadecimal conta de 0 a 15. Mas, como há somente dez dígitos (0 a 9), substitutos precisam ser encontrados para os números restantes.

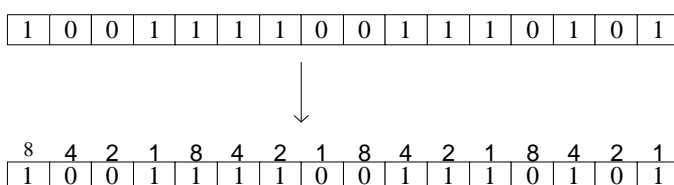
Por exemplo: 10, 11, 12, 13, 14 e 15. Os primeiros seis caracteres do alfabeto são usados como índices de reposição.

Por exemplo: A à F respectivamente.

Como resultado de contagem de base 16, 4 bits binários são necessários para representar uma base 16 ou número hexadecimal. Portanto, uma word de dados de 16 bits terá um código hexadecimal de 4 dígitos.

Há, na verdade, uma quarta interpretação para toda esta seqüência de bits. Esta é a leitura BCD ou Binary Coded Decimal (Decimal codificado binário). A seção a seguir converte o padrão de bits original num formato BCD.

a) Uma conversão BCD

Usando o padrão de bits original como base mas adicionando as seguintes identificações BCD permite a conversão de dados binários para o formato BCD.

Ao converter o valor no registrador acima para um valor BCD teremos o resultado igual à "ERRO".

Isto porque a conversão não será correta.

A razão é porque os números BCD só podem ter valores de 0 a 9, mas o segundo grupo de 4 bits da esquerda teria um valor igual à 14. Portanto um erro.

O processo de conversão é muito semelhante ao do hexadecimal exceto pelo limite mencionado dos valores de 0 a 9. Se os outros blocos fossem convertidos, somente como exemplo, os seguintes valores seriam encontrados;

Bloco do extremo lado esquerdo = ((1 x 8) + (1 x 1)) = 9

Segundo bloco do lado direito = ((1 x 4) + (1 x 2) + (1 x 1)) = 7

Bloco do extremo lado direito = ((1 x 4) + (1 x 1)) = 5

Dados BCD são lidos da esquerda para a direita como deve ser lido um número normal. Portanto, neste exemplo o "9" na verdade representaria "9000". O segundo bloco do lado direito é na verdade "70" e não, "7". As unidades são fornecidas pelo bloco do lado extremo direito, por exemplo, 5. As centenas "100s" teriam sido fornecidas pelo segundo bloco do lado esquerdo (o que é no exemplo possui um valor inválido). Também é importante notar que não há sinal com dados BCD convertidos. O número máximo permitido para uma única word de dados é "9999" e o mínimo é "0000".

Resumo dos dados de word

Em cada um dos casos anteriores o padrão de bits original tinha um outro significado. Para recapitular as três novas leituras e o padrão de bits original,

1	0	0	1	1	1	1	0	0	1	1	1	0	1	0	1

Decimal: -24971

Hexadecimal: 9E75

BCD: Erro (9?75)

Cada significado é radicalmente diferente do outro, mas todos são formas diferentes de descrever a mesma coisa. São todos, na verdade, iguais uns aos outros em diferentes bases de conversão.

2.14.4 COMPLEMENTO DE "2"

Controladores programáveis, computadores, etc., usam um formato chamado complemento de "2". Isto é um procedimento matemático que está adequado aos requerimentos de hardware operacional dos microprocessadores. É utilizado para representar números negativos e executar operações de subtração.

O procedimento é muito simples. No exemplo a seguir, os valores "15 e 7" vão esclarecer esta dúvida:

Passo1: Encontrar os valores binários (este exemplo usa 8 bits)

15 = 00001111

7 = 00000111

Passo 2: Encontra a inversão do valor a ser subtraído.

Procedimento: inverte todos os bits com valor "1" para o valor "0" e vice-versa.

7 = 00000111

7 invertido = 11111000

Passo 3: Adiciona 1 ao número invertido.

Procedimento: adicionar 1 ao bit menos significativo do byte. Lembrar que esta é uma adição binária, portanto, quando o valor 2 é obtido, 1 é movido para a próxima posição da esquerda e o restante é setado para 0 (zero);

7 invertido 11111000

1 adicional 00000001

Resposta 11111001

Este resultado é, na verdade, o mesmo que o valor negativo para 7, por exemplo. -7.

Passo 4: Adicione a resposta ao número com o qual está sendo feita a subtração (por exemplo. 15).

Procedimento: Lembrar que 1+1 = 0 levar 1 na base 2 (binário).

Valor original 15 00001111

Reposta encontrada no passo 3 11111001

Solução (1)00001000

O "(1)" é levado "1" e é ignorado porque este exemplo está lidando somente com 8 bits.

Passo 5: Converte a resposta de volta.

00001000 = 8

A resposta é positiva porque o MSB (o bit mais significativo) é um 0 (zero). Em caso de verificação mental rápida, verifica-se que o problema realmente encontrou que "15-7 = 8".

De fato, não houve nenhuma subtração. Cada um dos passos ou converteu alguns dados ou executou uma adição. No entanto, a resposta correta é 15 - 7 = 8. Este exemplo de cálculo foi baseado em números de 8 bits, mas funcionará da mesma forma em outra quantidade de bits.

2.15 NOTAÇÃO CIENTÍFICA E PONTO FLUTUANTE

PLCs podem utilizar vários sistemas e métodos diferentes para armazenar dados.

Os mais comuns já foram discutidos nas seções anteriores, por exemplo: BCD, Binário, Decimal, Hexadecimal, etc. Estes são conhecidos como formatos "inteiros" ou 'formatos de números inteiros'.

Como sugere o título, estes formatos usam somente números inteiros sem nenhuma representação de partes fracionárias. Entretanto, há dois outros formatos que estão se tornando cada vez mais importantes. Eles são:

a) Ponto Flutuante

b) Notação científica

Os dois formatos estão, na verdade, intimamente relacionados. Ambos se propõem a criar números muito grandes ou números muito pequenos que podem descrever componentes inteiros e fracionários.

Nota geral:

Por vezes as palavras 'Formato', 'Modo' e 'Notação' são trocadas quando as descrições destes processos numéricos são feitos. Entretanto, todas estas palavras fornecem o mesmo valor descritivo e, assim sendo, os usuários deveriam saber de suas existência.

Algumas constantes úteis

π	3.141×10^{0}
2π	6.283×10^{0}
$\pi/4$	7.853×10^{-1}
π^2	9.869×10^{0}
A velocidade da luz	2.997×10^{8} m/s
Gravidade, g	9.807×10^{0} m/s^2
e	2.718×10^{0}

Pontos fixos:

Ponto de ebulição de oxigênio líquido	-1.8297×10^{2} °C
Ponto de derretimento do gelo	0.00×10^{0} °C
Ponto triplo da água	1.00×10^{-2} °C
Ponto de ebulição	1.00×10^{2} °C

2.15.1 NOTAÇÃO CIENTÍFICA

Este formato poderia ser chamado o passo entre os formatos 'inteiros' e os formatos ponto flutuante. Em termos básicos, a notação científica usa dois dispositivos para armazenar informação sobre um número ou valor. Um endereço contém uma linha de dados dos caracteres num número (chamados de mantissa), enquanto que o segundo endereço contém informação sobre o número de casas decimais existem no número (chamado de expoente). Portanto, a notação científica pode acomodar valores maiores/menores que o limite normal de 32 bits. Por exemplo: -2.147.483.648 à 2.147.483.647 onde os limites da notação científica são:

Máximos	Mínimos
9999×10^{35}	9999×10^{-41}
-9999×10^{35}	-9999×10^{-41}

A notação científica pode ser obtida usando BCD, ou EBCD, instrução (FNC 18 ou FNC 118). Nesta situação, os números do formato do ponto flutuante são convertidos pela instrução BCD para notação científica.

A notação científica pode ser convertida de volta para o formato de ponto de flutuante usando a instrução BIN (FNC 19).

Os seguintes pontos deveriam ser lembrados sobre o uso da notação científica:

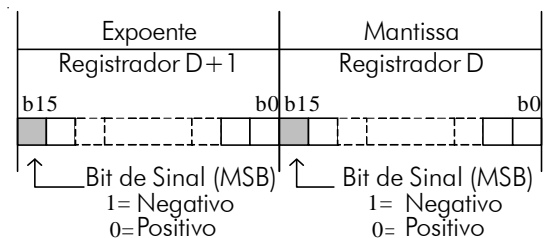
☑A mantissa e expoente estão armazenados em registradores de dados consecutivos.

Cada parte é composta por 16 bits e pode ser designado um valor positivo ou negativo indicado pelo valor do bit mais significante (MSB, ou bit 15 do registro de dados) para cada número.

☑ A mantissa está armazenada como os primeiros 4 números significantes sem arredondamento do número, por exemplo, o número de ponto flutuante valor 2.34567×10^{3} seria armazenado com uma mantissa de 2345 no registro de dados D e um expoente 0 (zero) no registro de dados D+1.

☑ A faixa de valores de mantissa disponível é 0, 1000 à 9999 e -1000 à -9999.

☑ A faixa de valores de expoente disponível é +35 até -41.

☑ Formato científico não pode ser usado diretamente em cálculos, mas fornece um método ideal para mostrar os dados numa interface de monitoramento.

2.15.2 FORMATO PONTO FLUTUANTE

O formato de ponto flutuante as possibilidades e faixas dadas pela notação científica podendo representar porções fracionárias de números inteiros. Por exemplo, executando e mostrando o cálculo de 22 dividido por 7 daria os seguintes resultados:

a) Operação normal de PLC usando números decimais (inteiros) resultaria em 3 sobrando 1
b) Em ponto flutuante, o resultado seria 3.14285 (aproximadamente)
c) No formato científico este cálculo seria igual a 3142×10^{-3}

Então, é possível ver o maior grau de precisão é dado pelos números em ponto flutuante. Por exemplo, através do uso de faixas numéricas maiores e a disponibilidade de mais dígitos calculáveis. Portanto, cálculos usando dados de ponto flutuante têm algumas vantagens significativas. Dados decimais podem ser convertidos para ponto flutuante usando a instrução FLT (FNC 49).

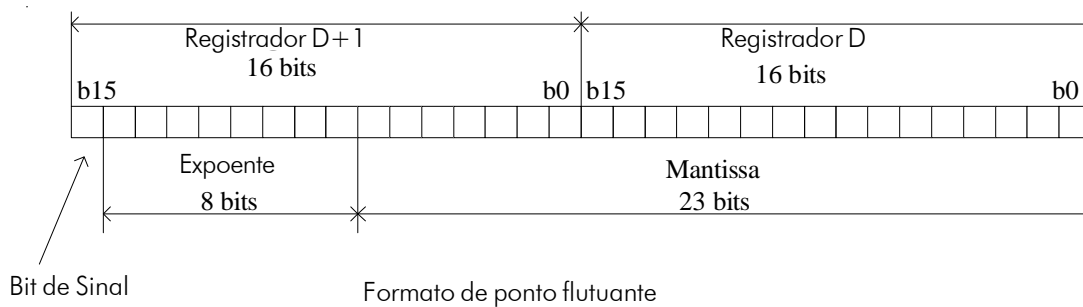
Os pontos a seguir deveriam ser lembrados sobre o uso do ponto flutuante;

☑ Números de ponto flutuante, não importando qual o valor numérico, sempre ocuparão dois registros de dados consecutivos (ou 32 bits).

☑ Valores de ponto flutuante não podem ser diretamente monitorados pois estão armazenados num formato especial recomendado pelo I.E.E.E (Institute of Electrical and Electronic Engineers – Instituto de Engenheiros Eletricistas e Eletrônicos) para aplicações pessoais e em micro computadores.

☑ Números em ponto flutuante têm tanto a mantissa quanto os expoentes (ver notação científica para uma explicação destes termos). No caso de expoentes de ponto flutuante, somente 8 bits são usados.

Adicionalmente, existe somente um sinal de bit para a mantissa. Os bits restantes do valor de 32 bits, por exemplo, 23 bits, são usados para 'descrever' o valor da mantissa.

Formato de ponto flutuante

Faixas válidas para uso de números de ponto flutuante:

Descrição	Sinais	expoente	Mantissa	Comentário
Ponto flutuante	0 ou 1	11111110 00000001	11111111111111111111111 11111111111111111111110 00000000000000000000001 00000000000000000000000	Número grande +/-3.403x10^{38} Precisão: 7 números significantes Menor número +/-1.175x10^{38}
Zero	0 ou 1	00000000	00000000000000000000000	Todos os dígitos são 0

CAPÍTULO 3

INSTRUÇÕES APLICÁVEIS

CAPÍTULO 3

INSTRUÇÕES APLICÁVEIS

INSTRUÇÕES APLICÁVEIS

CAPÍTULO 3

INSTRUÇÕES APLICÁVEIS

3 INSTRUÇÕES APLICÁVEIS

Instruções aplicadas são instruções especiais do PLC. Elas permitem ao usuário executar manipulações complexas de dados, operações matemáticas e ainda facilitam a programação e monitoração. Cada instrução aplicada tem um mnemônico único e números de função especiais. Cada instrução aplicada será mostrada usando uma tabela similar à que segue abaixo:

Mnemônico	Função	Operandos	Passos do programa
		D	
CJ FNC 00 (Salto condicional)	Executa um salto para uma posição associada por um label (ponteiro)	Ponteiros válidos da faixa de 0 a 255	CJ,CJP:3 passos Ponteiro do salto P□□1 passo

A tabela será encontrada no início de cada nova descrição de instrução. A área identificada como 'Operandos' dará uma lista de vários endereços (operandos) que podem ser usados com a instrução.

Várias letras de identificação serão usadas para associar cada operando com sua função, ex: destino, S - origem, n, m - número de elementos. Sufixos numéricos adicionais serão anexados se houver mais de um operando com a mesma função.

Nenhuma modificação do mnemônico de instrução é necessária para uma operação de 16 bits. Entretanto, operação habilitadas por pulsos requer que um 'P' seja adicionado diretamente após o mnemônico enquanto que a operação de 32 bits requer que um 'D' seja adicionado antes do mnemônico. Isto significa que se uma instrução estava sendo usada com os dois modos, pulsos e a operação de 32 bits, a identificação seria D□□□P onde □□□ era o mnemônico básico.

A função de 'pulso' permite que a instrução associada seja ativada na borda de subida da entrada de controle. A instrução será habilitada durante um ciclo varredura do programa.

Portanto, mesmo que a entrada de controle permanece setada, a instrução associada não estará ativa.

Para re-executar a instrução a entrada de controle deve sofrer nova transição positiva.

A seção MARCADORES identifica quaisquer marcadores que são usados pela instrução. Detalhes sobre a função do marcador são explicados no texto de instruções.

☑ Para instruções que operam continuamente, ex: a cada varredura do programa, a instrução operará e fornecerá um resultado novo diferente, o seguinte símbolo de identificação será usado '★' para representar um estado de mudança de alta velocidade. Instruções típicas cobertas por esta situação possuem um forte índice de incremento à sua operação.

☑ Na maioria dos casos, os operandos de instruções podem ser indexados pelo programa do usuário. Para aqueles operandos que não podem ser indexados, foi usado o símbolo '☆' que significa um operando inalterado depois ter sido escrito.

☑ Certas instruções utilizam registros de dados adicionais e/ou marcadores de status, por exemplo, uma função matemática como ADD (FNC 20) pode identificar um resultado zero, tomar emprestado e carry condições usando os marcadores auxiliares pré-definidos, como M8020 à M8021 respectivamente.

3.1 FLUXO DO PROGRAMA-FUNÇÕES 00 À 09

Conteúdo:

CJ -	Salto condicional	FNC 00
CALL -	Chamada subrotina	FNC 01
SRET -	Retorno de subrotina	FNC 02
IRET -	Retorno de uma interrupção	FNC 03
EI -	Habilitar interrupção	FNC 04
DI -	Desabilitar interrupção	FNC 05
FEND -	Primeiro Final	FNC 06
WDT -	Temporizador Watchdog	FNC 07
FOR -	Início do Loop	FNC 08
NEXT -	Retorno do Loop	FNC 09

Lista de símbolos:

D – Endereço de destino.

S – Endereço de origem.

m, n - Número de endereços ativos, bits ou uma constante operacional.

Sufixos numéricos adicionais serão anexados se houver mais de um operando com a mesma função, ex: D1, S3 ou para endereços de lista/tabelados D3+0, S+9 etc.

MSB – Bit mais significante, por vezes usado para indicar o sinal matemático de um número, ex: positivo = 0, e negativo = 1.

LSB – Bit menos significante.

Modificações de instrução:

☐☐☐ - Instrução de 16 bits, onde ☐☐☐ identifica a instrução mnemônica.

☐☐☐P - Instrução de 16 bits habilitada por pulso (simples).

D☐☐☐ - Instrução de 32 bits.

D☐☐☐P - Instrução de 32 bits habilitada por pulso (simples).

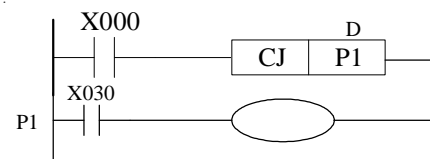
★ - Uma instrução repetitiva que mudará o valor de destino a cada varredura menos quando for habilitada por um pulso.

☆ - Um operando que não pode ser indexado, ex: A adição de V ou Z ou é inválida ou não terá efeito para o valor do operando.

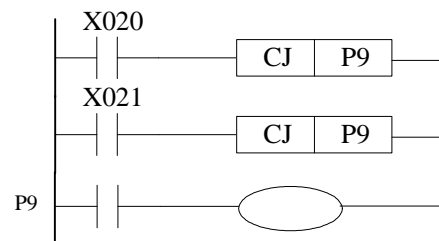
3.1.1 CJ (FNC 00)

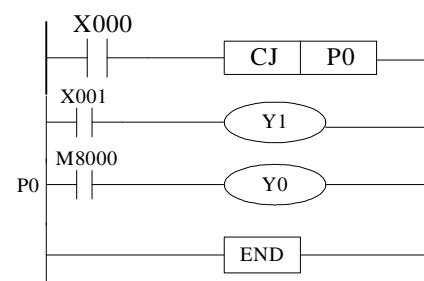
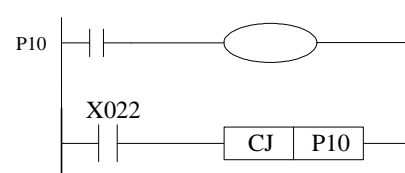
Mnemônico	Função	Operandos D	Passos do programa
CJ FNC 00 (Salto condicional)	Salta para um label (ponteiro) associado a instrução	Ponteiros válidos da faixa 0 a 255	CJ,CJP:3 passos ponteiro de salto P☐☐:1 passo

Operação:

Quando a instrução CJ está ativa ele força o programa a pular para um ponto do programa identificado por um ponteiro associado. Enquanto o salto acontece os passos de programa entre o início e o fim do salto não são executados. Isto significa que eles não são processados de maneira nenhuma. O efeito resultante é aumentar a velocidade do tempo de varredura operacional dos programas.

Pontos a observar:

a) Muitas instruções CJ podem referenciar um único ponteiro.

b) Cada ponteiro tem que ter um número exclusivo. Usando o ponteiro P63 é equivalente a pular para a instrução END.

c) As áreas do programa que não forem executadas por causa do salto, não atualizarão o status de saída mesmo que os dispositivos entrada mudem.

Por exemplo, o programa ao lado mostra uma situação que X001 aciona saída Y1. Presumindo que X001 estava energizada quando a instrução CJ foi ativada, alinha de comando onde está X001 e Y1 não será executada, mantendo Y1 energizada. Agora, mesmo que X001 esteja desenergizada, Y1 permanecerá energizada enquanto a instrução CJ força o programa a pular para o ponteiro P0. A situação inversa também se aplica, se X001 está desenergizada antes da instrução CJ ser acionada, Y1 será energizada mesmo que X001 seja setada. Uma vez que a instrução CJ é desativada, X001 acionará Y1 da maneira normal. Esta situação se aplica para todos os tipos de saídas, ex: Endereços SET, RST, OUT, Y, M e S

d) A instrução CJ pode pular para qualquer ponto dentro do programa principal e após uma instrução FEND.

e) A instrução CJ pode ser usada para pular adiante no programa, em direção à instrução END ou ela pode pular para trás em direção ao passo 0. Se um salto para trás for usado, deve-se tomar cuidado para evitar que o programa entre em um looping e o temporizador watchdog seja setado, pois o PLC entrará numa situação de erro.

f) Saltos incondicionais podem ser inseridos usando marcadores auxiliares especiais como o M8000.

Nesta situação enquanto o PLC está em RUN, o programa SEMPRE executará a instrução CJ de forma incondicional.

IMPORTANTE:

☑ Temporizadores e contadores congelarão seus valores atuais se não forem executados por instruções de salto.

Por exemplo, se Y1 no programa anterior (ver ponto c) foi substituido por T0 K100 e a instrução CJ for acionada, o conteúdo de T0 não mudaria/aumentaria até que a instrução CJ não estivesse mais acionada, ex: o valor atual do temporizador congelaria.

Contadores de alta velocidade são a única exceção a esta situação, pois são processados de forma independente do programa principal.

Usando instruções aplicadas:

☑ Instruções aplicáveis também deixam de ser executadas forem programadas entre a instrução CJ e ponteiro de destino. Entretanto, as instruções PLSY (FNC 57) e PWM (FNC 58) operarão continuamente se elas foram ativadas e antes que da instrução CJ ser acionada, do contrário elas serão processadas como instruções aplicadas padrão.

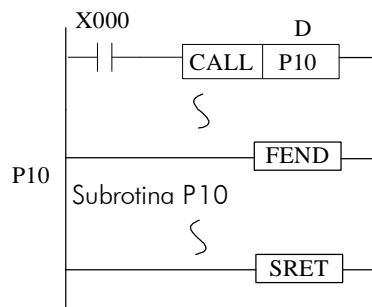
Detalhes de como usar a instrução CJ com outras instruções de fluxo de programa.

☑ Outros detalhes podem ser encontrados nas páginas 20 sobre o uso combinado de técnicas de fluxo de programa diferentes (como controle mestre, MC etc).

3.1.2 CALL (FNC 01)

Mnemônico	Função	Operandos D	Passos do programa
CALL FNC 01 (Chamada de Subrotina)	Executa a chamada de subrotina iniciando na posição do ponteiro associado	Ponteiros válidos da faixa 0 a 255 Níveis de aninhamento: 16, incluíndo a chamada inicial	CALL, CALLP: 3 passos Ponteiro de subrotina P□□: 1 passo

Operação:

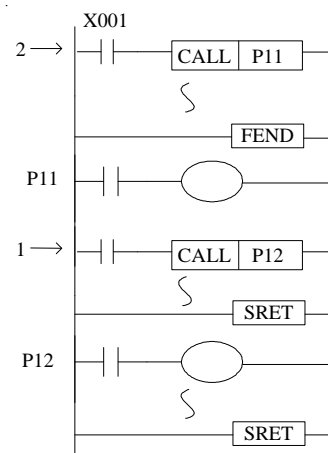
Quando a instrução CALL está ativa ela força o programa a executar a subrotina associada ao ponteiro (área identificada como subrotina P10). Uma instrução CALL deve ser usada em conjunto com instruções FEND (FNC 06) e SRET (FNC 02). O programa pula para o ponteiro de subrotina (localizado depois da instrução FEND) e processa o conteúdo até que uma instrução SRET seja encontrada. Isto força o fluxo do programa a voltar para a linha da lógica ladder imediatamente após a instrução CALL original.

Pontos a observar:

a) Diversas instruções CALL podem se referenciar a mesma subrotina.

b) Cada subrotina deve ter um único número de ponteiro. Ponteiros de subrotina podem ser selecionados de uma faixa P0 a P255. Ponteiros de subrotina e os ponteiros usados para instruções CJ (FNC 00) NÃO podem coincidir.

c) Subrotinas não são normalmente processadas, pois ocorrem após uma instrução FEND. Quando elas são chamadas, deve-se tomar cuidado para que o programa não entre num looping e o temporizador de watchdog seja setado.

d) Subrotinas podem ser aninhadas em até 16 níveis, considerando a instrução CALL inicial. Como exemplo, o programa à direita mostra um anihamento de 2 níveis.

Quando X001 está ativado o programa chama a subrotina P11. Nesta subrotina há um CALL para uma segunda subrotina, P12. Quando as duas subrotinas, P11 e P12 são ativadas simultaneamente, diz-se que estão aninhadas. Uma vez que a subrotina P12 alcança sua instrução SRET ela retorna ao programa ao passo imediatamente após a instrução CALL que a originou (ver 1→). O P11 então completa sua operação, e uma vez que sua instrução SRET é processada, o programa retorna mais uma vez para o passo que segue a chamada da instrução CALL P11 (ver 2→).

3.1.3 SRET (FNC 02)

Mnemônico	Função	Operandos D	Passos do programa
SRET FNC 02 (Retorno da Sub-rotina)	Retorna ao passo após a instrução CALL que habilitou a sub-rotina	N/A Retorna automaticamente ao passo que segue imediatamente após a instrução CALL que ativou a subrotina	SRET:1 passo

Operação:

SRET significa o fim da subrotina atual e retorna o fluxo do programa ao passo que imediatamente segue a instrução CALL que ativou a subrotina.

Pontos a observar:

a) O SRET só pode ser usado com a instrução CALL.

b) O SRET é sempre programado após uma instrução FEND – ver a instrução CALL (FNC 01) para mais detalhes.

3.1.4 IRET, EI, DI (FNC, 02, 04, 05)

Mnemônico	Função	Operandos D	Passos do programa
IRET FNC 03 (Retorno de interrupção)	Finaliza a execução de uma rotina habilitada por uma interrupção.	N/A Retorna automaticamente para o passo do programa principal que estava sendo processado no momento da chamada da interrupção.	IRET: 1 passo
EI FNC 04 (Habilita interrupções)	Possibilita que entradas de interrupção sejam processadas	N/A Qualquer entrada de interrupção sendo ativada após a instrução EI e antes das instruções FEND ou DI serão processadas imediatamente a menos que tenham sido especificamente desabilitadas.	EI: 1 passo
DI FNC 05 (Desabilita interrupções)	Desabilita o processamento de rotinas de interrupção	N/A Qualquer entrada de interrupção sendo ativada após uma instrução DI e antes de uma instrução EI será armazenada até que a próxima instrução EI seqüencial seja processada.	DI: 1 passo
I (Ponteiro de interrupção)	Identifica o início de uma rotina de interrupção	Um código numérico de 3 dígitos associado ao tipo de interrupção e operação	I□□□: 1 passo

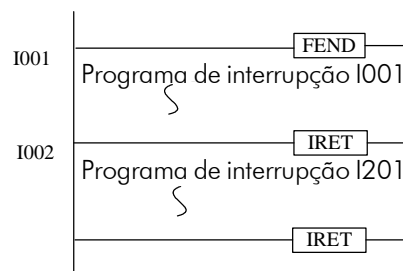
Descrição geral de uma rotina de interrupção:

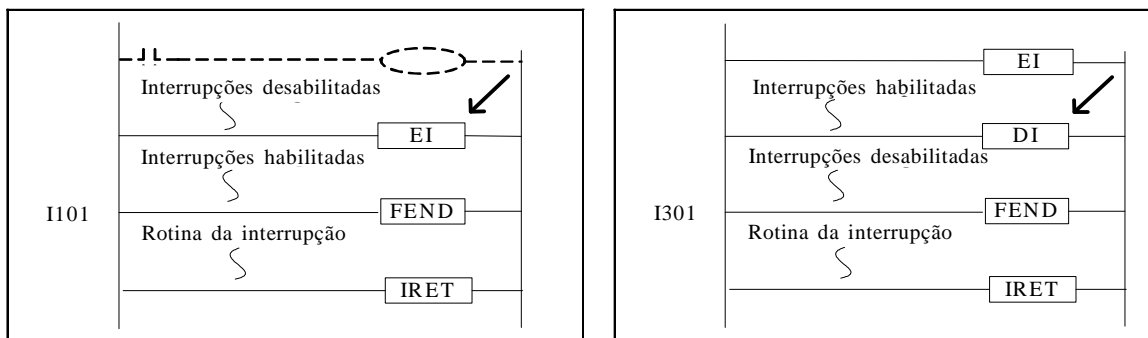
Uma rotina de interrupção é uma seção de programa que, quando habilitada, é imediatamente executada interrompendo o fluxo de programa principal. Uma vez que a interrupção foi processada o fluxo de programa principal continua de onde estava antes de ocorrer a interrupção.

Operação:

Interrupções são habilitadas por condições de entrada diferentes, às vezes uma entrada direta como a X0 é usada e outras vezes um intervalo de tempo de 30 mseg, pode ser usado. Programar e operar rotinas de interrupção requer até 3 instruções dedicadas (aquelas detalhadas nesta seção) e um ponteiro de interrupção.

Definindo uma rotina de interrupção:

Uma rotina de interrupção é especificada entre seu exclusivo ponteiro de interrupção e a primeira ocorrência de uma instrução IRET. Rotinas de interrupção são SEMPRE programadas após uma instrução FEND. A instrução IRET só pode ser usada em rotinas de interrupção.

Controlando operações de interrupção:

O PLC possui um status padrão para desabilitar a operação de interrupção. A instrução EI deve ser usada para ativar a execução das interrupções. Todas as interrupções físicas que ocorrem durante o período de varredura do programa desde a instrução EI até as instruções FEND ou DI terão suas rotinas de interrupção associada executadas. Se estas interrupções são habilitadas fora da faixa de habilitação (EI-FEND ou EI-DI, ver diagrama abaixo) elas serão armazenadas até que a instrução EI seja processada na varredura seguinte. Neste momento, a rotina de interrupção será executada.

Se uma interrupção individual tiver que desabilitar seu marcador especial M associado n deverá estar setado. Enquanto esta bobina está habilitada, a rotina de interrupção não será ativada. Para detalhes sobre como desabilitar interrupções com bobinas do tipo M, ver as tabelas de endereços do PLC no capítulo 5.

Alinhamento de Interrupções:

Interrupções podem ser aninhadas em até dois níveis. Isto significa que uma interrupção pode ser interrompida durante sua operação. Entretanto, para conseguir isto a rotina de interrupção que poderá ser outra vez interrompida deve conter as instruções EI e DI; caso contrário, como numa operação normal, quando a rotina de interrupção é ativada todas as outras interrupções são desabilitadas.

Interrupções de ocorrência simultânea:

Se mais que uma interrupção ocorre sequencialmente, é dada prioridade à interrupção que está ocorrendo primeiro. Se duas ou mais interrupções ocorrem simultaneamente, é dada alta prioridade à rotina de interrupção com o número de ponteiro mais baixo.

Usando temporizadores gerais em rotinas de interrupção:

O PLC tem uma faixa de temporizadores especiais que podem ser usados em rotinas de interrupção, para maiores informações ver item 2.9.3.

Sinais de trigger de entrada – duração de pulso:

Rotinas de interrupção que são habilitadas diretamente pelas entradas de interrupção, como a X0 etc., requerem uma duração de sinal de aproximadamente 20μ seg, por exemplo, a largura do pulso de entrada é igual a ou maior que 200μ seg. Quando este tipo de interrupção é selecionada, os filtros de entrada do hardware são automaticamente definidos em 50μ seg. (sob circunstâncias normais de operação os filtros de entrada são definidos em10mseg.).

Função de reconhecimento de pulso de curta duração

Entradas de alta velocidade diretas podem ser usadas para reconhecer sinais pulsados de curta duração. Quando um pulso é recebido numa entrada o marcador M especial correspondente será habilitado. Isto permite que o pulso lido seja usado para disparar outras ações no programa, mesmo que o sinal original esteja agora desenergizado. O PLC requer a instrução EI (FNC 04) para ativar o reconhecimento de um pulso nas entradas X0 até X5. Notar que, se um endereço de entrada está sendo usado por outra função de alta velocidade, então o reconhecimento deste pulso para aquele endereço será desabilitado.

3.1.5 FEND (FNC 06)

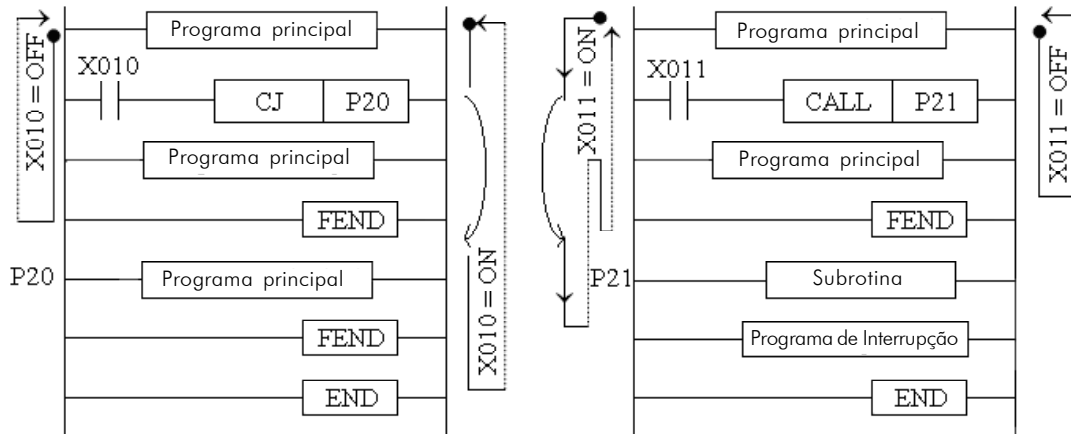
Mnemônico	Função	Operandos	Passos do programa
		D	
FEND FNC 06 (Fim do programa principal)	Usado para indicar o final do bloco de programa principal	N/A Nota: Pode ser usado com CJ (FNC 00), CALL (FNC 01) e as rotinas de interrupção	FEND: 1 passo

Operação:

Uma instrução FEND indica o final do programa principal e o início da área de programa a ser usado por subrotinas. Sob circunstâncias normais de operação a instrução FEND executa uma ação similar a da instrução END, por exemplo, processamento de saída, atualização das entradas e zerando o temporizador de watchdog.

Pontos a observar:

a) A instrução FEND é de forma geral usada só com construções de programa CJ-P-FEND, CALL-P-SRET e I-IRET (P refere-se ao ponteiro do programa, I refere-se ao ponteiro de interrupção).

Os dois ponteiros/subrotinas CALL e subrotinas de ponteiros de interrupção (I) são SEMPRE programados após uma instrução FEND, ex: estas características do programa NUNCA aparecem no corpo do programa principal.

a) Ocorrências múltiplas de instruções FEND podem ser usadas para separar subrotinas diferentes (ver diagrama acima).

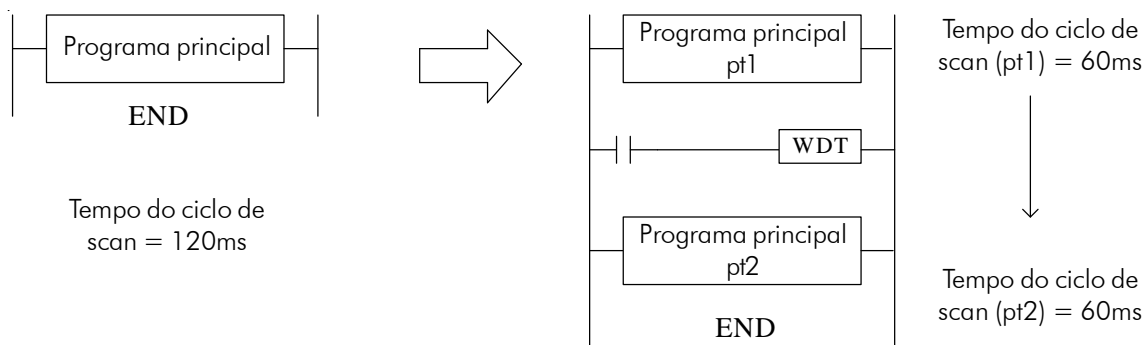
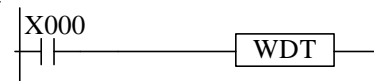
b) As construções de fluxo de programa NÃO podem ser separadas por uma instrução FEND.

c) FEND nunca pode ser usada após uma instrução END.

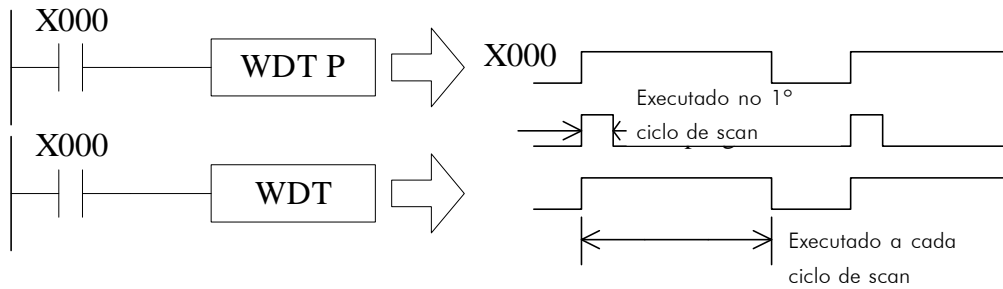
3.1.6 WDT (FNC 07)

Mnemônico	Função	Operandos D	Passos do programa
WDT FNC 07 (Zera o Temporizador Watch dog)	Usado para zerar o temporizador watch dog durante uma varredura de programa	N/A Pode ser acionado a qualquer momento dentro do corpo do programa principal	WDT, WDTP: 1 passo

Operação:

A instrução WDT zera o temporizador de watchdog do PLC. O temporizador de watchdog verifica se o tempo de varredura do programa (operação) não excedeu um limite de tempo pré-definido. Presume-se que, se este limite de tempo é excedido haverá, em algum momento, um erro na execução do programa. O PLC então parará a execução do programa do usuário para evitar a ocorrência de quaisquer outros erros. Fazendo com que o temporizador de watchdog seja zerado (acionando a instrução WDT) o limite de tempo de varredura (operação de programa) é efetivamente aumentado.

Pontos a observar:

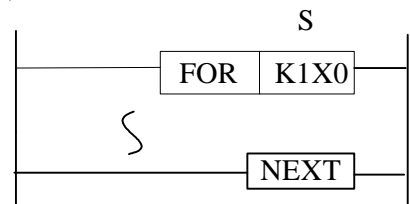
a) Quando a instrução WDT é usada ela operará em cada varredura de programa contanto que sua condição de entrada tenha sido satisfeita.

Para forçar uma instrução WDT a operar por somente UMA varredura, o usuário precisa programar algum tipo de intertravamento. Usuários têm opção de usar o formato de pulso (P), ex.: WDTP.

b) O temporizador de watchdog tem parametrização padrão de 200 mseg. Este limite de tempo pode ser ajustado conforme o conteúdo do registro de dados D8000, o registro do temporizador watchdog.

3.1.7 FOR, NEXT (FNC 08, 09)

Mnemônico	Função	Operandos D	Passos do programa
FOR FNC 08 (Início do looping FOR-NEXT)	Identifica a posição inicial e o número de repetições para o loop	K, H, KnX, KnY, KnM, KnS, T, C, D, V, Z	FOR: 3 passos
NEXT FNC 09 (Fim do looping FOR-NEXT)	Identifica a posição final do loop	N/A Nota: O loop FOR-NEXT pode ter até 16 níveis de aninhamento,por exemplo, 16 loops FOR-NEXT são programados um no outro	NEXT: 1 passo

Operação:

As instruções FOR e NEXT permitem a repetição especifica de uma área de programa, por exemplo, o programa anexado pelas instruções que serão repetidas S número de vezes.

Pontos a observar:

a) A instrução FOR opera num modo de 16 bits, portanto, o valor do operando S pode setar dentro de uma faixa de 1 a 32,767. Se um número na faixa entre -32,768 e 0 (zero) é especificado, ele é automaticamente substituído pelo valor 1, por exemplo, o loop FOR-NEXT executaria apenas uma vez.

b) A instrução NEXT NÃO tem operando.

c) As instruções FOR-NEXT devem ser programadas como um par, ou seja, para cada instrução FOR é NECESSÁRIA uma instrução NEXT associada. O mesmo se aplica às instruções NEXT, é NECESSÁRIA uma instrução FOR associada. As instruções FOR-NEXT também devem ser programadas na ordem correta. Isto significa que programar um loop como NEXT-FOR (a instrução NEXT precede a instrução FOR associada) NÃO é permitido.

Inserir uma instrução FEND entre as instruções FOR-NEXT, como FOR-FEND-NEXT, NÃO é permitido. Isto teria o mesmo efeito que programar uma instrução FOR sem uma NEXT, seguida da instrução FEND e um loop com um NEXT e nenhuma instrução associada FOR.

d) Um loop FOR-NEXT opera pelo número de vezes programado antes que o programa principal possa terminar a varredura atual do programa.

e) Ao usar loops FOR-NEXT deve-se cuidar em não exceder o limite do temporizador watchdog do PLC. O uso da instrução WDT e/ou incrementar o valor de temporização do temporizador watchdog é recomendado.

Aninhamento de Loops FOR-NEXT:

Instruções FOR-NEXT podem ser aninhadas em até 16 níveis. Isto significa que 16 loops FOR-NEXT podem ser sequencialmente programados um ao outro.

No exemplo, um aninhamento de 3 níveis foi programado. À medida que cada novo nível de aninhamento FOR-NEXT é encontrado, o número de vezes que aquele loop é repitido é aumentado pela multiplicação de todos os loops anteriores.

Por exemplo, o loop C opera 4 vezes. Mas dentro deste loop há um loop B aninhado. Para cada ciclo de loop completo de C, o loop B será completamente executado, ou seja, ele será executado D0Z vezes. Isto também se aplica entre os loops B e A.

O número total de vezes que o loop A operará para UMA varredura do programa será igual a;

1) O número de operações do loop A multiplicado pelo

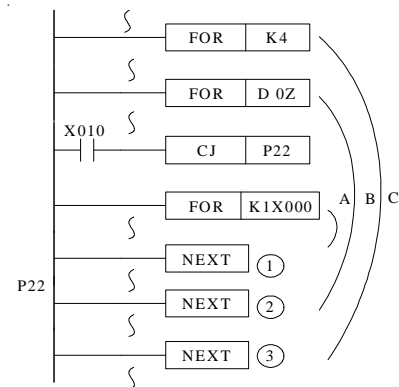
2) Número de operações do loop B multiplicado pelo

3) Número de operaçãos do loop C

Se fossem associados valores aos loops A, B e C, como 7, 6 e 4 respectivamente, o seguinte número de operações aconteceria em UMA varredura de programa:

Número de operações do loop C = 4 vezes

Número de operações do loop B = 24 vezes (C x B, 4 x 6)

Número de operações do loop A = 168 vezes (C x B x A, 4 x 6 x 7)

Nota:

O uso da característica de programação CJ, que causa o pulo para P22 permite a 'seleção' de qual loop será processado e quando, por exemplo, se X10 estiver setada, o loop A já não operaria.

3.2 MOVER E COMPARAR - FUNÇÕES 10 À 19

Conteúdo:

CMP -	Comparar	FNC 10
ZCP -	Comparar área de memória	FNC 11
MOV -	Mover	FNC 12
SMOV -	Shift Mover	FNC 13
CML -	Complemento	FNC 14
BMOV -	Mover Bloco	FNC 15
FMOV -	Preenche e Move	FNC 16
XCH -	Troca	FNC 17
BCD -	Decimal Codificado Binário	FNC 18
BIN -	Binário	FNC 19

Lista de símbolos:

D – Endereço de destino.

S – Endereço de origem.

m, n - Número de endereços ativos, bits ou uma constante operacional.

Sufixos numéricos adicionais serão anexados se houver mais de um operando com a mesma função, ex: D1, S3 ou para dispositivos de lista/tabelados D3+0, S+9 etc.

MSB –Bit mais significante, por vezes usado para indicar o sinal matemático de um número, ex: positivo = 0, e negativo = 1.

LSB – Bit menos significante.

Modificações de instrução:

☐☐☐- Instrução de 16 bits, onde ☐☐☐ identifica a instrução mnemônica.

☐☐☐P - Instrução de 16 bits habilitada por pulso (simples).

D☐☐☐ - Instrução de 32 bits.

D☐☐☐P - Instrução de 32 bits habilitada por pulso (simples).

★ -Uma instrução repetitiva que mudará o valor de destino a cada varredura menos quando for habilitada por um pulso.

☆ - Um operando que não pode ser indexado, ex: A adição de V ou Z ou é inválida ou não terá efeito para o valor do operando.

3.2.1 CMP (FNC 10)

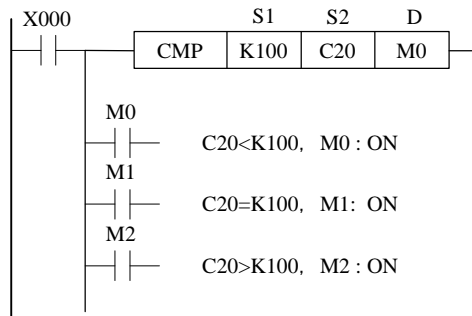
Mnemônico	Função	Operandos			Passos do programa
		S1	S2	D	
CMP FNC 10 (Comparação)	Compara dois endereços informando se o resultado é <, = ou >	K, H, KnX, KnY, KnM, KnS, T, C, D, V, Z		Y,M,S Nota: 3 endereços consecutivos são usados	CMP, CMPP: 7 passos DCMP, DCMPP: 13 passos

Operação:

Os dados de S1 são comparados aos dados de S2. O resultado é indicado por 3 bits especificos do endereço principal digitado como D. Os endereços de bits indicam:

S2 é menor que S1 – endereço de bits D é setado

S2 é igual a S1 - endereço de bits D+1 é setado

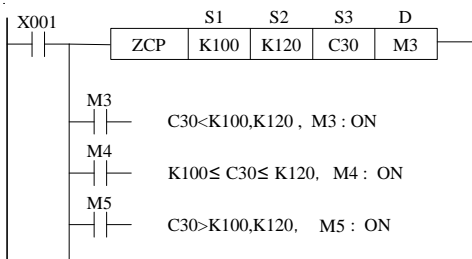
S2 é maior que S1 – endereço de bits D+2 é setado

Nota: O status do endereço de destino (D) será mantido mesmo que a instrução CMP seja desativada. Comparações algébricas inteiras são usadas, ex: -10 é menor que +2 etc.

3.2.2 ZCP (FNC 11)

Mnemônico	Função	Operandos				Passos do programa
		S1	S2	S3	D	
ZCP FNC 11 (Comparação de área de memória)	Compara o valor de dados numa faixa de endereços informando se o resultado foi <, = ou >.	K, H, KnX, KnY, KnM, KnS, T, C, D, V, Z Nota: S1 deverá ser menor que S2			Y, M, S Nota: 3 endereços consecutivos são usados.	ZCP, ZCPP: 9 passos DZCP,DZCPP: 17 passos

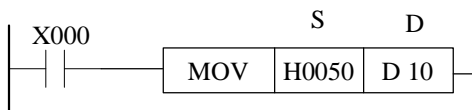
Operação:

A operação é a mesma que para a instrução CMP exceto que um único valor de dados (S3) é comparado na faixa de dados (S1-S2). S3 é menor que S1e S2 – o endereço de bits D é setado. S3 é igual ou está entre S1 e S2 – endereço de bits D+1 é setado

S3 é maior que S1 e S2 – endereço de bits D+2 é setado.

3.2.3 MOV (FNC 12)

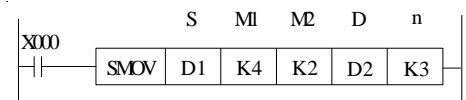
Mnemônico	Função	Operandos		Passos do programa
		S	D	
MOV FNC 12 (Mover)	Move dados de uma área de memória para uma nova área	K, H, KnX, KnY, KnM, KnS, T, C, D, V, Z	KnY, KnM, KnS, T, C, D, V, Z	MOV, MOVP: 5 passos DMOV, DMOVP: 9 passos

Operação:

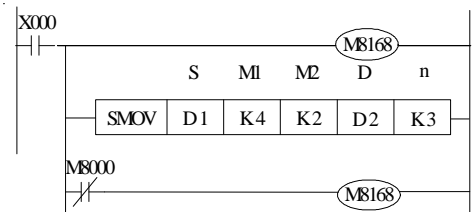
O conteúdo do endereço de origem (S) é copiado para o endereço de destino (D) quando a entrada de controle está ativa. Se a instrução MOV não está acionada, não há operação nenhuma.

3.2.4 SMOV (FNC 13)

Mnemônico	Função	Operandos					Passos do programa
		M1	M2	N	S	D	
SMOV FNC 13 (Shift mover)	Move um valor BCD de um endereço de origem para um endereço de destino	K, H Nota: faixa disponível 1 a 4 ✰			K, H, KnX, KnY, KnM, KnS, T,C,D,V,Z	K, H, KnY, KnM, KnS, T,C,D,V,Z	SMOV, SMOVP: 11 passos
					Faixa de 0 a 9,999 (decimal) ou 0 a 9,999 (BCD) quando o M8168 estiver setado		

Operação 1:

Esta instrução copia um número especificado de dígitos de um endereço de origem decimal de 4 dígitos (S) e coloca-os num lugar específico dentro de um destino (D) número (também um decimal de 4 dígitos). Os dados existentes no destino serão sobrescritos.

m1 – A posição na origem do primeiro dígito a ser movido
m2 – O número de dígitos na origem a serem movidos
n- A posição de offset no destino para o primeiro dígito
Nota: O destino selecionado NÃO deve ser menor que a quantidade de dados de fonte.
Posições de dígitos são referenciadas por número: 1= unidades, 2= dezenas, 3= centenas, 4=milhares.

Operação 2:

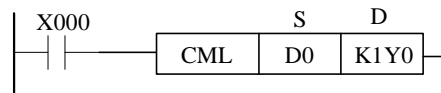
Esta instrução SMOV permite que números BCD sejam manipulados exatamente da mesma forma que o SMOV 'normal' manipula números decimais, por exemplo, esta instrução copia um número especifico de dígitos de uma fonte BCD de 4 dígitos (S) e coloca-os numa localização especificada dentro de um número de destino (D) (também um número BCD de 4 dígitos).Para selecionar o modo BCD a instrução SMOV é associada à um marcador especial M8168 que deve ser setado. Lembrar que esta é uma operação de parametrização do 'modo' da instrução e estará ativa, ou seja, todas as instruções SMOV operarão em formato BCD até que o modo seja zerado, ex: M8168 desabilitado.

3.2.5 CML (FNC 14)

Mnemônico	Função	Operandos		Passos do programa
		S	D	
CML FNC 14 (Complemento)	Copia e inverte o padrão de bits do endereço de origem para um destino especifico	K, H, KnX, KnY, KnM, KnS, T, C, D, V, Z	KnY, KnM, KnS, T, C, D, V, Z	CML,CMLP: 5 passos DCML,DCMLP: 9 passos

Operação:

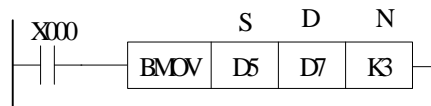
Uma cópia de cada bit de dados no endereço de origem (S) é invertido e depois movido para um destino designado (D).

X000 S D
CML D0 K1Y0

Isto significa que cada ocorrência de '1' na origem tornar-se-á '0' nos dados de destino enquanto cada dígito que for '0' tornar-se-á '1'. Se a área de destino é menor que os dados da origem então somente os endereços de bit mapeados diretamente serão processados.

3.2.6 BMOV (FNC 15)

Mnemônico	Função	Operandos			Passos do programa
		S	D	N	
BMOV FNC 15 (Movimentação de bloco)	Move um determinado bloco de endereços para uma nova área de memória	KnX, KnY, KnM, KnS, T,C,D, V, Z	KnY, KnM,KnS, T, C, D, V, Z	K, H, D ☆ Nota: N ≤ 512	BMOV, BMOVP: 7 passos

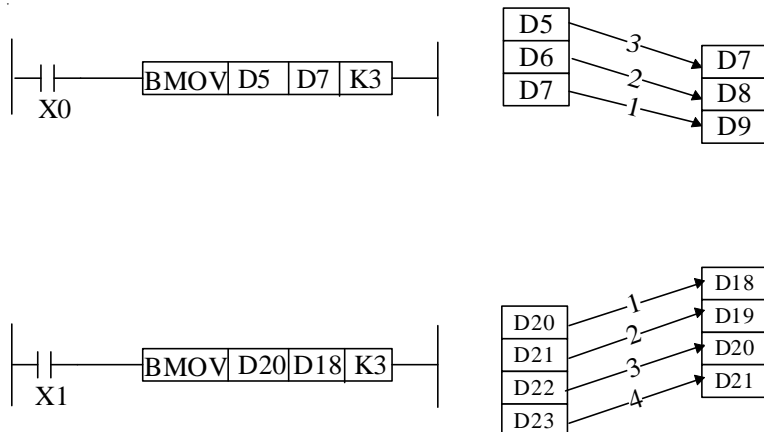
Operação:

Uma quantidade de dados consecutivos pode ser copiada para uma nova área de memória. Os dados de origem são identificados como um endereço inicial (S) e a quantidade de dados consecutivos (n). O bloco de dados é movido para o endereço de destino (D) com o mesmo número de elementos (n).

Pontos a observar:

a) Se a quantidade de endereços fonte (n) exceder o número real de endereços disponíveis, então, somente aqueles endereços que se enquadram na faixa disponível serão usados.

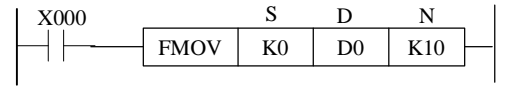
b) Se o número de dados na origem exceder o espaço disponível na área de destino, então somente se escreverá a quantidade de dados que couberem na área disponível no destino.

c) A instrução BMOV possue uma característica automática interna para prevenir que se escrevam dados errados na ocorrência das faixas de dados de fonte (S - n) e destino (D -n) coincidentes. Isto está claramente identificado no diagrama a seguir:

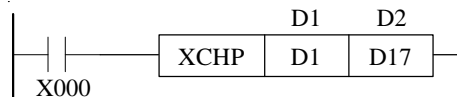
Nota: As flechas numeradas indicam a ordem na qual o BMOV é processado

3.2.7 FMOV (FNC 16)

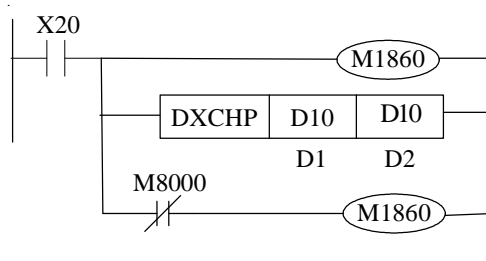
Mnemônico	Função	Operando			Passos do programa
		S	D	N	
FMOV FNC 16 (Fill move)	Copia um único dado num determinado endereço para uma área de memória	KnX, KnY, KnM, KnS, T, C, D, V, Z	KnY, KnM, KnS, T, C, D, V, Z	K, H ☆ Nota: N ≤ 512	FMOV, FMOVP: 7 passos DFMOV, DFMOVP: 13 passos

Operação:

Os dados armazenados em um endereço de origem (S) são copiados diversos endereços de destino. O primeiro endereço a ser escrito é definido pelo parâmetro (D) e a quantidade de endereços consecutivos a serem escritos por (n). Se o número de endereços a serem afetados pela instrução (n) exceder o espaço disponível na área de memória de destino, então somente os endereços disponíveis serão alterados.

3.2.8 XCH (FNC 17)

Mnemônico	Função	Operandos		Passos do programa
		D1	D2	
XCH FNC 17 (Troca) ★	Troca as posição de dados entre os endereços definidos	KnY, KnM, KnS, T, C, D, V, Z Nota: Quando operando com byte XCH (M8160 setado) D1 e D2 devem ser o mesmo endereço, caso contrário um erro de programa ocorrerá e o M8067 será setado.		XCH, XCHP: 5 passos DXCH, DXCHP: 9 passos

Operação 1: O conteúdo dos dois endereços D1 e D2 são trocados, ou seja, o valor que estava em D1 irá para D2 e vice-versa. Ex.

Registradores	Antes XCH	Depois XCH
D1	20	530
D17	530	20

Operação 2: Esta função é equivalente ao FNC 147 SWAP, os bytes em cada word dos endereços designados D1 são trocados quando o marcador M8160 é setado. Ex.

Registradores		Antes DXCH	Depois DXCH
D10	Byte 1	1FH	8BH
	Byte 2	8BH	1FH
D11	Byte 1	C4H	35H
	Byte 2	35H	C4H

3.2.9 BCD (FNC 18)

Mnemônico	Função	Operandos		Passos do programa
		S	D	
BCD FNC 18 (Decimal codificado binário)	Converte números binários para BCD	KnX,KnY, KnM, KnS, T, C, D, V, Z	KnY, KnM, KnS, T, C, D, V, Z	BCD, BCDP: 5 passos DBCD, DBCDP: 9 passos

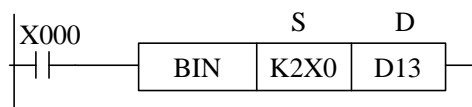
Operação:

Os dados de origem binária (S) são convertidos para o formato BCD e armazenados no endereço de destino (D). Se o número BCD convertido excede as faixas operacionas de 0 a 9.999 (operação de 16 bits) e 0 a 99.999.999 (operação de 32 bits) ocorrerá erro. Esta instrução pode ser usada diretamente para dados de saída como apresentação de um dado num display de 7 segmentos.

3.2.10 BIN (FNC 19)

Mnemônico	Função	Operandos		Passos do programa
		S	D	
BIN FNC 19 (Binário)	Converte números BCD em seus equivalentes binários.	KnX,KnY, KnM, KnS, T, C, D, V, Z	KnY, KnM, KnS, T, C, D, V, Z	BIN, BINP: 5 passos DBIN, DBINP: 9 passos

Operação:

Os dados de origem BCD (S) são convertidos em número binário equivalente e armazenados no endereço de destino (D). Se os dados de origem não são fornecidos num formato BCD ocorrerá erro. Esta instrução pode ser usada para ler dados diretamente de chaves thumbwheel.

3.3 OPERAÇÕES ARITMÉTICAS E LÓGICAS - FUNÇÕES 20 À 29

Conteúdo:

ADD -	Adição	FNC 20
SUB -	Subtração	FNC 21
MUL -	Multiplicação	FNC 22
DIV -	Divisão	FNC 23
INC -	Incrementa	FNC 24
DEC -	Decrementa	FNC 25
WAND -	Operação lógica 'E'	FNC 26
WOR -	Operação lógica 'OU'	FNC 27
WXOR -	Op. lógica 'OU Exclusivo'	FNC 28
NEG -	Inversão	FNC 29

Lista de símbolos:

D – Endereço de destino.

S – Endereço de origem.

m, n - Número de endereços ativos, bits ou uma constante operacional.

Sufixos numéricos adicionais serão anexados se houver mais de um operando com a mesma função, ex: D1, S3 ou para endereços de lista/tabelados D3+0, S+9 etc.

MSB – Bit mais significante, por vezes usado para indicar o sinal matemático de um número, ex: positivo = 0, e negativo = 1.

LSB – Bit menos significante.

Modificações de instrução:

☐☐☐- Instrução de 16 bits, onde ☐☐☐ identifica a instrução mnemônica.

☐☐☐P - Instrução de 16 bits habilitada por pulso (simples).

D☐☐☐ - Instrução de 32 bits.

D☐☐☐P - Instrução de 32 bits habilitada por pulso (simples).

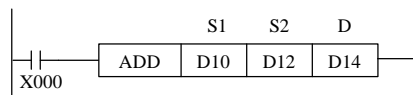
★ - Uma instrução repetitiva que mudará o valor de destino a cada varredura menos quando for habilitada por um pulso.

☆ - Um operando que não pode ser indexado, ex: A adição de V ou Z ou é inválida ou não terá efeito para o valor do operando.

3.3.1 ADD (FNC 20)

Mnemônico	Função	Operandos			Passos do programa
		S1	S2	D	
ADD FNC 20 (Adição)	Os valores dos dois endereços são somados e o resultado é armazenado no endereço de destino	K, H, KnX, KnY, KnM, KnS,T, C, D, V, Z		KnY, KnM, KnS, T, C, D, V, Z	ADD, ADDP: 7 passos DADD, DADDP: 13 passos

Operação:

Os dados contidos nos endereços de origem (S1, S2) são combinados e o resultado é armazenado no endereço de destino especificado (D).

Pontos a observar:

a) Os cálculos são processados de forma algébrica, ex: 5 + (-8)= -3.

b) O mesmo endereço pode ser usado como origem (S1 ou S2) e como destino (D). Se este é o caso então a instrução ADD na verdade operaria continuamente. Isto significa que em cada varredura a instrução adicionaria o resultado da última varredura ao penúltimo endereço de origem. Para evitar que isto aconteça habilitação por pulso deveria ser usada ou um intertravamento deveria ser programado.

c) Se o resultado de um cálculo é "0" então um marcador auxiliar M8020 é setado.

d) Se o resultado de uma operação excede 32.767 (limite de 16 bits) ou 2.147.483.647 (limite de 32 bits) o bit de carry, M8022 é setado. Se o resultado de uma operação excede -32.768 ou -2.147.483.648 o bit de borrow, M8021 é setado. Quando o resultado excede quaisquer dos limites numéricos, o marcador apropriado será setado (M8021 ou M8022) e o bit de carry/borrow será armazenado no endereço de destino. O sinal matemático destes dados armazenados reflete no limite do número que foi excedido, ex: quando -32.768 é excedido, números negativos são armazenados no endereço de destino, mas se 32.767 foi excedido, números positivos seriam armazenados em D.

e) Se o local de destino não comportar o tamanho do dado a ser armazenado, então somente a porção do resultado que couber na área de destino será escrita, por exemplo, se 25 (decimal) foi o resultado, e seria armazenado em K1Y4 então somente Y4 e Y7 estariam ativos. Em termos binários isto é o equivalente a um valor decimal de 9, muito longe do resultado real de 25!

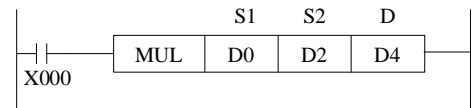
3.3.2 SUB (FNC 21)

Mnemônico	Função	Operandos		Passos do programa
		S1, S2	D	
SUB FNC 21 (Subtração)	Os valores dos dois endereços são subtraidos e o resultado é armazenado no endereço de destino	K, H, KnX, KnY, KnM, KnS, T, C, D, V, Z	KnY, KnM, KnS, T, C, D, V, Z	SUB, SUBP: 7 passos DSUB, SUBP: 13 passos

Operação: Os dados contidos no endereço de origem S2 é subtraído do conteúdo de S1. O resultado é armazenado no endereço de destino D. Nota: os 'Pontos a observar', abaixo da instrução ADD (na página anterior) também podem ser aplicados de forma similar à instrução de subtração.

3.3.3 MUL (FNC 22)

Mnemônico	Função	Operandos			Passos do programa
		S1	S2	D	
MUL FNC 22 (Multiplicação)	Os valores dos dois endereços são multipicados e o resultado é armazenado no endereço de destino	K, H, KnX, KnY, KnM, KnS, T, C, D, V, Z		KnY,KnM,KnS, T, C, D, Z(V) Note: Z(V) não pode ser usado para operações de 32 bits.	MUL, MULP: 7passos DMUL, DMULP: 13 passos

Operação: O conteúdo dos dois endereços de origem (S1, S2) é multiplicado e o resultado é armazenado no endereço de destino (D). Note que se aplicam as regras normais de álgebra.

Pontos a observar:

a) Ao executar a instrução MUL no modo 16 bits, dois endereços de16 bits são multiplicados. Eles produziram um resultado de 32 bits, o endereço identificado como destino é o menor de dois endereços usados para armazenar o resultado de 32 bits. Ao usar o exemplo acima com dados de teste, verificou-se:

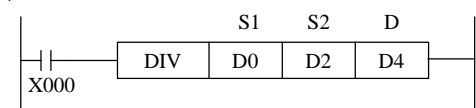
5 (D0) x 7 (D2) = 35 - O valor 35 é armazenado em (D4, D5) como uma double word de 32 bits simples.

b) Ao executar a instrução MUL no modo 32 bits, dois endereços de 32 bits são multiplicados. Eles produziram um resultado de 64 bits. O endereço editado no campo (D) será o offset dos quatro endereços de memória que armazenaram o resultado de 64 bits.

c) Se a área de memória disponível no destino não for sufiente para armazenar o resultado obtido, então somente o valor que couber na área disponível será armazenado, por exemplo, se o resultado for 72 (decimal) seria armazenado em K1Y4 então somente Y7 estaria ativo. Em termos binários isto é o equivalente a um valor decimal de 8, muito longe do resultado real de 72!

3.3.4 DIV (FNC 23)

Mnemônico	Função	Operandos			Passos do programa
		S1	S2	D	
DIV FNC 23 (Divisão)	Os valores dos dois endereços são divididos e o resultado é armazenado no endereço de destino	K, H, KnX, KnY, KnM, KnS, T, C, D, V, Z		KnY,KnM,KnS, T, C, D, Z(V) Note: Z(V) não podem ser utilizados em operações 32 bits	DIV, DIVP: 7 passos DDIV, DDIVP: 13 passos

Operação: O conteúdo do endereço (S1) é dividido pelo conteúdo de (S2). O resultado é armazenado no destino (D). Notar que se aplicam as regras normais de álgebra.

Pontos a observar:

a) Ao executar a instrução DIV em16 bits, duas fontes de dados de16 bits são divididas uma pela outra. Elas produzem dois resultados de 16 bits. O offset do endereço de destino será o valor editado no campo (D).

Este endereço de armazenagem, na verdade, terá um registro do número de quantas vezes S2 se dividirá em S1 (o quociente).

O segundo registro de destino contém o que resta após a divisão completa (o resto). Usando o exemplo anterior com alguns dados de testes, temos:

51 (D0) ÷ 10 (D2) = 5(D4) 1(D5)

Este resultado é interpretado como 5 divisões completas com resto 1 (5 x10 + 1 = 51).

b) Ao executar a instrução DIV em 32 bits, o conteúdo de dois endereços de 32 bits é dividido. Ele produzirá dois resultados de 32 bits. O primeiro endereço de destino é o editado no campo (D) da instrução, sendo que o primeiro armazenará o quociente e os dois dispositivos seguintes serão usados para armazenar o que restou. Se D30 foi selecionado como o destino de uma operação de divisão de 32 bits, então D30 e D31 armazenariam o quociente e D32 e D33 armazenariam o resto. Se a área de memória disponível no destino não for sufiente para armazenar o resultado obtido, então somente o valor que couber na área disponível será armazenado. Se endereços de bits são usados como área de destino, nenhum valor restante é calculado.

c) Se o valor do endereço de origem S2 é 0 (zero), então é gerado um erro de operação e a operação da instrução DIV é cancelada.

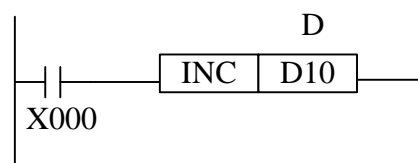
3.3.5 INC (FNC 24)

Mnemônico	Função	Operandos D	Passos do programa
INC FNC 24 (Incrementa) ★	A cada execução da instrução o conteúdo do endereço definido irá incrementar 1	KnY, KnM, KnS, T, C, D, V, Z Para V e Z operando em 32 bits ver item 2.13	INC, INCP: 3 passos DINC, DINCP: 5 passos

Operação: Em cada execução da instrução, o conteúdo do endereço especificado no campo D será incrementado (acrescentado) pelo valor de 1.

Na operação de 16 bits, quando +32.767 é alcançado o próximo incremento escreverá um valor de -32.768 no endereço de destino.

Na operação de 32 bits, quando +2.147.483.647 é alcançado o próximo incremento escreverá um valor de -2.147.483.648 no endereço de destino.

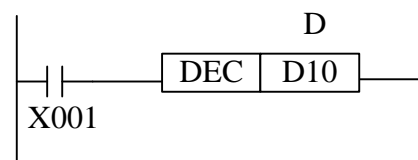
Nos dois casos não há marcadores adicionais para identificar esta mudança no valor contado.

3.3.6 DEC (FNC 25)

Mnemônico	Função	Operandos D	Passos do programa
DEC FNC 25 (Decrementa) ★	A cada execução da instrução o conteúdo do endereço definido irá decrementar 1	KnY, KnM, KnS, T, C, D, V, Z Para V e Z operando em 32 bits ver item 2.13	DEC, DECP: 3 passos DDEC, DDECP: 5 passos

Operação: Em cada execução da instrução, o conteúdo do endereço especificado no campo D será decrementado (subtraído) pelo valor de 1.

Na operação de 16 bits, quando -32.767 é alcançado, o próximo decremento escreverá um valor de -32.768 no endereço de destino.

Na operação de 32 bits, quando -2.147.483.647 é alcançado, o próximo decremento escreverá um valor de -2.147.483.648 no endereço de destino.

Nos dois casos não há marcadores adicionais para identificar esta mudança no valor contado.

3.3.7 WAND (FNC 26)

Mnemônico	Função	Operandos			Passos do programa
		S1	S2	D	
WAND FNC 26 (Operação lógica "E")	Uma operação AND é executada entre os endereços de origem e o resultado armazenado no destino	K, H, KnX, KnY, KnM, KnS, T, C, D, V, Z		KnY, KnM, KnS, T, C, D, V, Z	WAND, WANDP: 7 passos DAND, DANDP: 13 passos

Operação:

Os padrões de bits dos dois endereços de origem são analizados (o conteúdo de S2 é comparado com o conteúdo de S1). O resultado da análise AND lógica é armazenado no endereço de destino (D).

As seguintes regras são usadas para determiner o resultado de uma opereção AND lógica. Isto acontece para cada bit contido nos endereços de origem:

Regra geral: (S1) Bit n WAND (S2) Bit n = (D) Bit n

1 WAND 1 = 1   0 WAND 1 = 0

1 WAND 0 = 0   0 WAND 0 = 0

3.3.8 WXOR (FNC 28)

Mnemônico	Função	Operandos			Passos do programa
		S1	S2	D	
WXOR FNC 28 (Operação lógica OU exclusivo)	Uma operação OU exclusivo é executada entre os endereços de origem e o resultado armazenado no destino	K,H,KnX,KnY, KnM, KnS, T, C, D, V, Z		KnY, KnM, KnS, T, C, D, V, Z	WXOR, WXORP: 7 passos DXOR, DXORP: 13 passos

Operação:

Os padrões de bits dos dois endereços de origem são analizados (o conteúdo de S2 é comparado com o conteúdo de S1). O resultado da análise XOR lógica é armazenado no endereço de destino (D).

As seguintes regras são usadas para determinar o resultado de uma operação XOR lógica. Isto acontece para cada bit contido nos endereços de origem:

Regra geral: (S1)Bit n WXOR (S2)Bit n = (D)Bit n

1 WXOR 1 = 0   0 WXOR 1 = 1

1 WXOR 0 = 1   0 WXOR 0 = 0

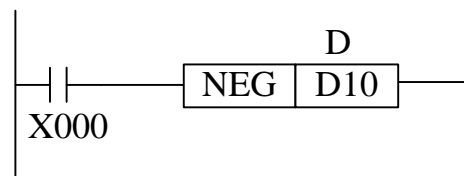
3.3.9 NEG (FNC 29)

Mnemônico	Função	Operandos			Passos do programa
		S1	S2	D	
NEG FNC 29 (Inversão) ★	Inverte de forma lógica o conteúdo do endereço designado	K,H, KnX,KnY, KnM, KnS, T, C, D, V, Z		KnY, KnM, KnS, T, C, D, V, Z	NEG, NEGP: 3 passos DNEG, DNEGP: 5 passos

Operação:

O padrão de bits do endereço selecionado é invertido.

Isto significa que qualquer ocorrência de ‘1’ torna-se um ‘0’ e qualquer ocorrência de ‘0’ será escrito como ‘1’.

Quando isto estiver completo, um outro 1 binário é adicionado ao padrão de bits. O resultado é a mudança lógica de sinal do conteúdo dos endereços selecionados, ou seja, um número positivo tornar-se á um número negativo ou vice versa.

3.4 ROTAÇÃO E SHIFT - FUNÇÕES 30 A 39

Conteúdo:

ROR -	Rotação Direita	FNC 30
ROL -	Rotação Esquerda	FNC 31
RCR -	Rotação Direita com Carry	FNC 32
RCL -	Rotação Esquerda com Carry	FNC 33
SFTR -	(Bit) Shift Direita	FNC 34
SFTL -	(Bit) Shift Esquerda	FNC 35
WSFR -	Word Shift Direita	FNC 36
WSFL -	Word Shift Esquerda	FNC 37
SFWR -	Shift Register Escrita	FNC 38
SFRD -	Shift Register Leitura	FNC 39

Lista de símbolos:

D – Endereços de destino.

S – Endereços de origem.

m, n - Número de endereços ativos, bits ou uma constante operacional.

Sufixos numéricos adicionais serão anexados se houver mais de um operando com a mesma função. Exemplo: D1, S3 ou para endereços como listas/tabelas D_{3+0}, S_{+9} etc.

MSB – Bit mais significante, às vezes usado para indicar o sinal matemático de um número. Por exemplo: positivo = 0, e negativo = 1.

LSB – Bit menos Significante.

Modificações da instrução:

☐☐☐ - Uma instrução operando no modo de 16 bits, ☐☐☐ identifica o mneumônico da instrução.

☐☐☐P - Uma instrução de modo de 16 bits modificada para usar operação de pulso (simples).

D☐☐☐ - Uma instrução modificada para operar numa operação de 32 bits.

D☐☐☐P - Uma instrução de modo de 32 bits modificada para usar operação de pulso (simples).

★ - Uma instrução repetitiva que mudará o valor do destino em cada varredura a menos que seja modificada pela função de pulso.

☆ - Um operando que não pode ser indexado. Por exemplo: a adição do valor de V ou Z é inválida ou não terá efeito para o valor do operando.

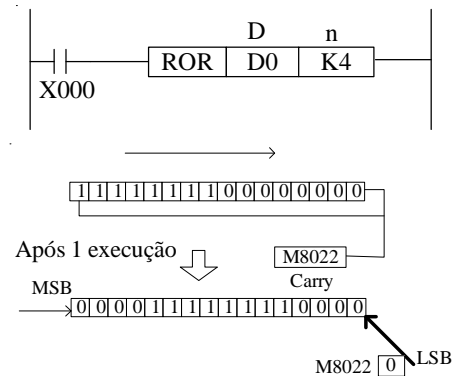
3.4.1 ROR (FNC 30)

Mnemônico	Função	Operandos		Passos do programa
		D	n	
ROR FNC 30 (Rotação direita) ★	O padrão de bits do endereço de destino é rotacionado 'n' posições para a direita em cada execução	KnY, KnM, KnS, T, C, D, V, Z Nota: Operação de 16 bits Kn=K4, Operação de 32 bits Kn=K8	K, H, ☆ Nota: Operação de 16 bits n? 16 Operação de 32 bits n≤32	ROR, RORP: 5 passos DROR, DRORP: 9 passos

Operação:

O conteúdo de bits do endereço de destino (D) é rotacionado "n" posições de bits para a direita em cada operação da instrução.

O status do último bit rotacionado é copiado para o marcador "carry" M8022.

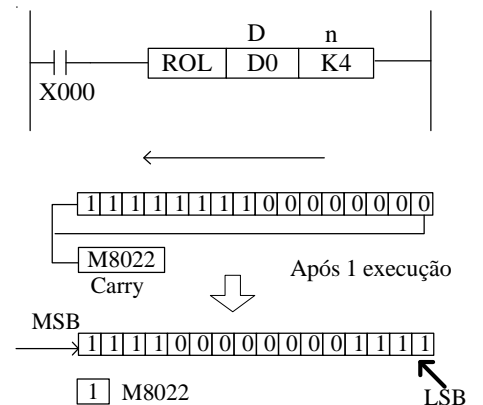
O exemplo mostrado à esquerda é baseado na instrução descrita acima onde o padrão de bits representa o conteúdo do D0.

3.4.2 ROL (FNC 31)

Mnemônico	Função	Operandos		Passos do programa
		S	D	
ROL FNC 31 (Rotação esquerda) ★	O padrão de bits do endereço de destino é rotacionado 'n' posições para a esquerda a cada execução	KnY, KnM, KnS, T, C, D, V, Z Nota: Operação de 16 bits Kn= K4, Operação de 32 bits Kn= K8	K, H, ☆ Nota: Operação de 16 bits n≤ 16 Operação de 32 bits n≤ 32	ROL, ROLP: 5 passos DROL, DROLP: 9 passos

Operação:

O conteúdo de bits do endereço de destino (D) é rotacionado "n" posições de bits para a esquerda em cada operação da instrução.

O status do último bit rotacionado é copiado para o marcador "carry" M8022.

O exemplo mostrado acima é baseado na instrução descrita acima onde o padrão de bits representa o conteúdo do D0.

3.4.3 RCR (FNC 32)

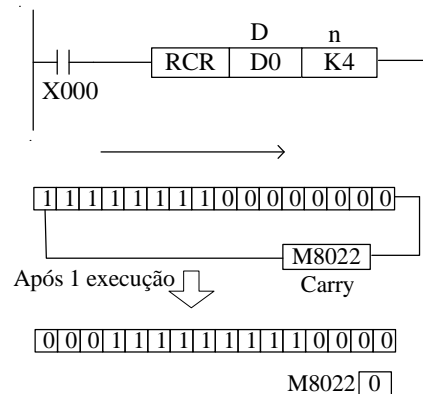
Mnemônico	Função	Operandos		Passos do programa
		D	n	
RCR FNC 32 (Rotação para direita com carry) ★	O conteúdo do endereço de destino é rotacionado para direita com 1 bit extraído do marcador "carry"	KnY, KnM, KnS, T, C, D, V, Z Nota: Operação de 16 bits Kn= K4, Operação de 32 bits Kn= K8	K, H, ☆ Nota: Operação de 16 bits n≤ 16 Operação de 32 bits n≤ 32	RCR,RCRP: 5 passos DRCR, DRCRP: 9 passos

Operação:

O conteúdo dos bits do endereço de destino (D) é rotacionado "n" posições de bits para a direita em cada operação da instrução.

O status do último bit rotacionado é copiado para o marcador "carry" M8022.

Na seguinte operação da instrução M8022 é o primeiro bit a ser levado de volta para o dispositivo de destino.

O exemplo mostrado à esquerda é baseado na instrução descrita acima onde o padrão de bits representa o conteúdo do D0.

3.4.4 RCL (FNC 33)

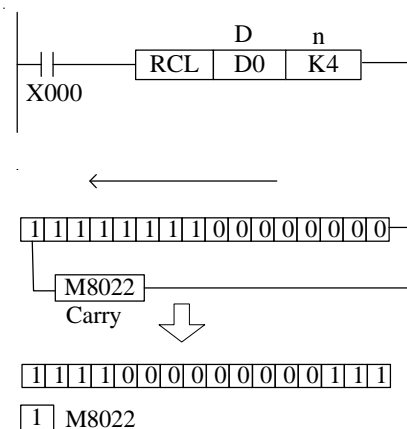
Mnemônico	Função	Operandos		Passos do programa
		S	D	
RCL FNC 33 (Rotação para esquerda com carry) ★	O conteúdo do endereço de destino é rotacionado para esquerda com 1 bit extraído do marcador "carry"	KnY, KnM, KnS, T, C, D, V, Z Nota: Operação de 16 bits Kn= K4, Operação de 32 bits Kn= K8	K, H, ☆ Nota: Operação de 16 bits n≤16 Operação de 32 bits n≤32	RCL, RCLP: 5 passos DRCL, DRCLP: 9 passos

Operação:

O conteúdo de bits do endereço de destino (D) é rotacionado "n" posições de bits para a esquerda em cada operação da instrução.

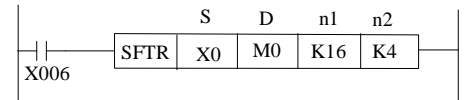
O status do último bit rotacionado é copiado para o marcador "carry" M8022.

Na seguinte operação da instrução, o estado de M8022 é o primeiro bit a ser levado de volta para o dispositivo de destino.

O exemplo mostrado à esquerda é baseado na instrução descrita acima onde o padrão de bits representa o conteúdo do D0.

3.4.5 SFTR (FNC 34)

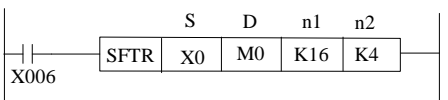
Mnemônico	Função	Operandos				Passos do programa
		S	D	n1	n2	
SFTR FNC 34 (Shift de bit para direita) ★	O conteúdo dos endereços fonte é copiado para uma pilha de bits levando os dados existentes para a direita	X, Y, M, S	Y, M, S	K, H, ☆ Nota: n2≤n1≤1024		SFTR,SFTRP: 9 passos

Operação:

A instrução copia o conteúdo do endereço fonte n2 para uma pilha de bits de comprimento n1. Para cada nova adição de bits n2, os dados existentes numa pilha de bits são movidos para n2 bits para a direita. Quaisquer dados de bits indo para uma posição excedendo o limite n1, são levados para uma área de "overflow". A operação de shifting de bits ocorrerá cada vez que a instrução é processada a menos que seja modificada ou com o sufixo de pulso ou um intertravamento controlado.

3.4.6 SFTL (FNC 35)

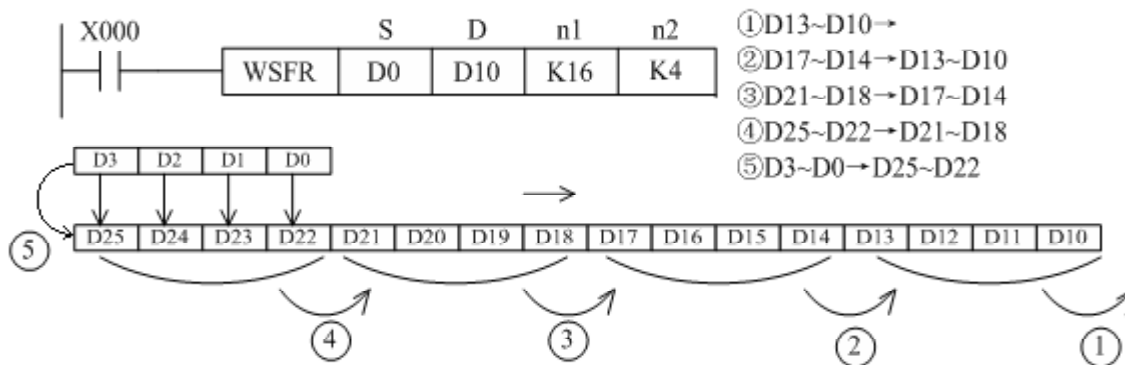
Mnemônico	Função	Operandos				Passos do programa
		S	D	N1	N2	
SFTL FNC 35 (shift de bit para esquerda) ★	O conteúdo dos endereços fonte é copiado para uma pilha de bits levando os dados existentes para esquerda	X, Y, M, S	Y, M, S	K, H, ☆ Nota: N2≤n1≤1024		SFTL,SFTLP: 9 passos

Operação:

A instrução copia o conteúdo do endereço fonte n2 para uma pilha de bits de comprimento n1. Para cada nova adição de bits n2, os dados existentes na pilha de bits são movidos para n2 bits para a direita. Quaisquer dados de bits indo para uma posição excedendo o limite n1 são levados para uma área de "overflow". A operação de shifting de bits ocorrerá cada vez que a instrução é processada a menos que seja modificada ou com o sufixo de pulso ou um intertravamento controlado.

3.4.7 WSFR (FNC 36)

Mnemônico	Função	Operandos				Passos do programa
		S	D	N1	N2	
WSFR FNC 36 (shift de word para direita) ★	O conteúdo dos endereços fonte é copiado para uma pilha de words levando os dados existentes para a direita	KnX, KnY, KnM,KnS, T, C, D	KnY, KnM,KnS T, C, D	K, H, ☆ Nota: n2 ≤n1 ≤512		SFTR,SFTRP: 9 passos

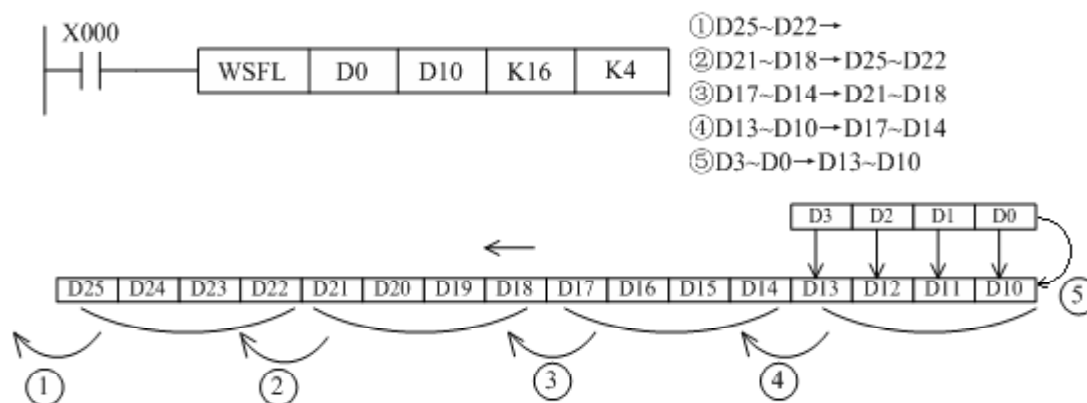
Operação:

A instrução copia o conteúdo do endereço n2 para uma pilha de word no comprimento n1. Para cada adição de words n2, os dados existentes na pilha de word é movido n2 words para a direita. Quaisquer dados de word indo para uma posição excedendo o limite n1 é levado para uma área de "overflow". A operação de shifting de word ocorrerá cada vez que a instrução é processada a menos que seja modificada ou com o sufixo de pulso ou um intertravamento controlado.

Nota: Ao usar endereços de bits como fonte (S) e destino (D) o valor Kn deve ser igual.

3.4.8 WSFL (FNC 37)

Mnemônico	Função	Operandos				Passos do programa
		S	D	N1	N2	
WSFL FNC 37 (Shift de word para esquerda) ★	O conteúdo dos endereços fonte é copiado para uma pilha de words levando os dados existentes para a esquerda	KnX, KnY, KnM,KnS, T, C, D	KnY,KnM, KnS, T, C, D	K, H, ☆ Nota: N2≤n1≤512		WSFL, WSFLP: 9 passos

Operação:

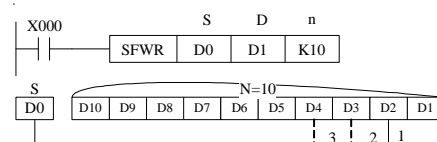
A instrução cópia o conteúdo do endereço fonte n2 para uma pilha de word de comprimento n1. Para cada adição de words n2, os dados existentes na pilha de word é movido n2 words para a esquerda. Quaisquer dados indo para uma posição excedendo o limite n1 são levados para uma área de "overflow". A operação de shifting de word ocorrerá cada vez que a instrução é processada a menos que seja modificada ou com o sufixo de pulso ou um intertravamento controlado.

Nota: Ao usar endereços de bits como fonte (S) e destino (D) o valor Kn deve ser igual.

3.4.9 SFWR (FNC 38)

Mnemônico	Função	Operandos			Passos do programa
		S	D	N	
SFWR FNC 38 (Shift register escrita) ★	Esta instrução cria e constrói uma pilha FIFO do tamanho do endereço n – deve ser usado com SFRD FNC 39	K, H, KnX, KnY, KnM,KnS, T, C, D, V, Z	KnY, KnM, KnS, T, C, D,	K, H, ☆ Nota: 2≤n1≤512	SFWR, SFWRP: 7 passos

Operação:

O conteúdo do endereço fonte (S) é escrito na pilha FIFO. A posição da inserção na pilha é automaticamente calculada pelo PLC.

O endereço de destino (D) é o principal endereço da pilha FIFO. O conteúdo de D identifica onde o próximo os registradores serão armazenados (com um offset de D+1). Se o conteúdo de D exceder o valor "n-1" (n é o comprimento da pilha FIFO) então a inserção dos dados na pilha FIFO é interrompida. O marcador "carry" M8022 é setado para identificar esta situação.

Pontos a observar:

a) FIFO é uma abreviação de 'First-In/ First-OUT' (Primeiro que entra, primeiro que sai).

b) Apesar dos endereços n serem associados à pilha FIFO, somente n-1 pedaços da informação podem ser escritos naquela pilha. Isto porque endereço principal (D) toma o primeiro registro disponível para armazenar a informação que diz respeito ao próximo ponto de inserção de dados na pilha FIFO.

c) Antes de começar a usar a pilha FIFO certifique-se que o conteúdo do registrador principal (D) é igual a '0' (zero).

d) Esta instrução deveria ser usada em conjunto com o SFRD FNC 39. O parâmetro n nas duas instruções deve ser igual.

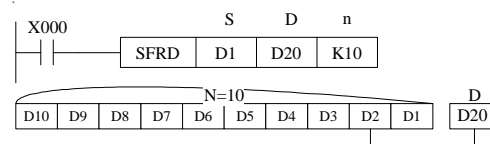
3.4.10 SFRD (FNC 39)

Mnemônico	Função	Operandos			Passos do programa
		S	D	N	
SFRD FNC 39 (Shift register leitura) ★	Esta instrução lê os dados armazenados na pilha FIFO - deve ser usado com SFWR FNC 38	KnY, KnM, KnS, T, C, D,	KnY, KnM, KnS, T, C, D, V, Z	K, H, ☆ Nota: 2 ≤ n ≤ 512	SFRD, SFRDP: 7 passos

Operação:

O endereço fonte (S) identifica o endereço principal da pilha FIFO. Seu conteúdo reflete o último ponto de entrada de dados na pilha FIFO. Por exemplo: onde fica o final da pilha FIFO (posição atual).

Esta instrução lê o primeiro pacote de dados da pilha FIFO (registro S+1), leva todos os dados na pilha, uma posição para "cima" para preencher a área lida e remover o conteúdo do endereço principal da pilha FIFO (S) em 1. Os dados lidos são escritos no endereço de destino (D).

Quando o conteúdo do endereço fonte (S) é igual a '0' (zero), por exemplo: a pilha FIFO está vazia e o marcador M8020 é setado.

Pontos a observar:

a) FIFO é uma abreviação de 'First-In/ First-OUT' (Primeiro que entra, primeiro que sai).

b) Somente n-1 pacotes de dados podem ser lidos naquela pilha. Isto porque a pilha necessita que o primeiro registro, o endereço principal (S) seja usado para conter informação sobre o comprimento atual da pilha FIFO.

c) Esta instrução sempre lerá os dados de fonte do registrador S+1.

d) Esta instrução deveria ser usada em conjunto com o SFWR FNC 38. O parâmetro n nas duas instruções deve ser igual.

3.5 OPERAÇÃO DE DADOS – FUNÇÕES 40 À 49

Conteúdo:

ZRST -	Zeramento de área de memória	FNC 40
DECO -	Decodificar	FNC 41
ENCO -	Codificar	FNC 42
SUM -	A soma de Bits ativos	FNC 43
BON -	Verificar Status de Bits específicos	FNC 44
MEAN -	Média	FNC 45
ANS -	(Cronometrado) Habilita diagnóstico	FNC 46
ANR -	Desabilita diagnósitico	FNC 47
SQR -	Raiz quadrada	FNC 48
FLT -	Ponto Flutuante	FNC 49

Lista de símbolos:

D – Endereço de destinato.

S – Endereço fonte.

m, n - Número de endereços ativos, bits ou uma constante operacional.

Suxifos numéricos podem ser adiocionados se houver mais do que um operando com a mesma função. Por exemplo: D1, S3 ou para endereços de listas/tabelas D3+0, S+9 etc.

MSB – Bit mais significante, às vezes usado para indicar o sinal matemático de um número, por exemplo: positivo = 0 ou negativo = 1.

LSB – Bit menos significante.

Modificações de instrução:

☐☐☐ - Uma instrução de 16 bits, onde ☐☐☐ identifica a instrução mnemônica.

☐☐☐P – Instrução de16 bits modificada para usar operação de pulso (simples).

D☐☐☐ – Instrução de 32 bits.

D☐☐☐P – Instrução de 32 bits modificada para usar operação de pulso (simples).

★ - Uma instrução repetitiva que mudará o valor do destino em cada varredura a menos que seja modificada pela função de pulso.

☆ - Um operando que não pode ser indexado, por exemplo: a adição de V ou Z ou é inválida ou não terá efeito no valor do operando.

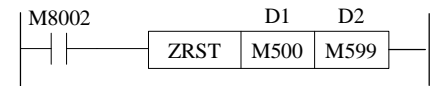
3.5.1 ZRST (FNC 40)

Mnemônico	Função	Operandos		Passos do programa
		D1	D2	
ZRST FNC 40 (Zeramento de uma área de memória)	Usado para zerar uma faixa de endereços semelhantes em uma única operação	Y, M,S, T, C, D Nota: D1 deve ser menor que ou igual (≤) a D2. Contadores padrão e de alta velocidade não podem ser misturados		ZRST, ZRSTP: 5 passos

Operação:

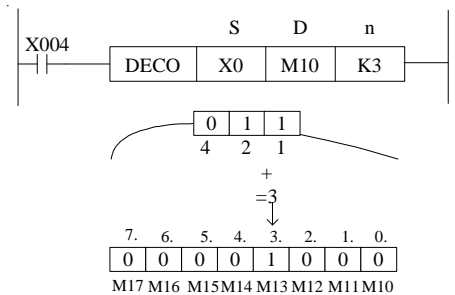
A faixa de endereços, inclusive daqueles especificados como os dois destinos são zerados, por exemplo: para endereços de word o valor atual será 0 (zero) e para endereços de bits, os status serão forçados para OFF.

A faixa de endereços não pode conter tipos de memória diferentes: por exemplo, o C000 especificado como o endereço inicial de destino (D1) não pode fazer estar na mesma instrução com o T199 como segundo endereço de destino (D2). Ao zerar contadores, os contadores padrão e de alta velocidade não podem ser zerados como fazendo parte da mesma faixa de endereços.

Se o D1 é maior que (>) D2, então somente o endereço D1 é zerado.

3.5.2 DECO (FNC 41)

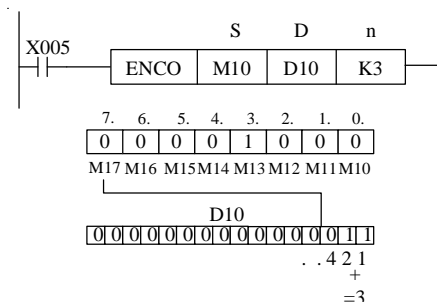
Mnemônico	Função	Operandos			Passos do programa
		S	D	N	
DECO FNC 41 (Decodificação)	O valor de dados de origem Q identifica o bit na posição Q do endereço de destino que setado.	K, H, X, Y, M,S, T, C, D, V,Z	Y, M, S, T, C, D	K, H, ☆Nota: D= Y,M,S então a faixa n = 1-8 D= T,C,D então a faixa n = 1-4 n = 0, então sem processamento	DECO, DECOP: 7 passos

Operação:

Os dados de origem são fornecidos por uma combinação dos operandos S e n. Onde S especifica o endereço principal dos dados e n, o número de bits consecutivos. Os dados de fonte são lidos com um único número (binário a conversão decimal) Q. O número fonte Q é a localização de um bit dentro do endereço destino (D) que será habilitado (ver exemplo ao lado). Quando o endereço de destino é um endereço de dados, o n deve estar dentro de uma faixa 1 a 4 já que só há 16 bits de destino disponíveis numa única word de dados. Todos os bits de dados não usados na word serão programados para 0.

3.5.3 ENCO (FNC 42)

Mnemônico	Função	Operandos			Passos do programa
		S	D	N	
ENCO FNC 42 (Encode)	A localização do bit ativo mais elevado é armazanada como uma posição numérica a partir do endereço principal	X, Y, M, S, T, C, D, V, Z	T, C, D, V,Z	K, H, ☆ Nota: S=X, Y, M, S então a faixa n =1-8 S= T,C,D então a faixa n = 1-4 n = 0, então sem processamento:	ENCO, ENCOP 7 passos

Operação:

O bit ativo mais alto dentro de uma faixa de leitura tem sua localização anotada com sendo um offset do endereço principal fonte (S). Isto é armazanado no resgistro de destino (D).

Pontos a observar:

a) A faixa de leitura é definida pelo maior número que pode ser armazenado num formato binário dentro da área de armazenamento destino especificada por n, por exemplo: se n fosse igual a 4 bits um número máximo na faixa de 0 à 15 pode ser escrito no endereço de destino. Portanto, se endereços de memória de bits estivessem sendo usados como dados de origem, endereços de 16 bits seriam usados para armazenamento, por exemplo: o endereço de bits principal e outros 15 endereços consecutivos.

b) Se o número de destino armazenado é 0 (zero) então o bit do endereço principal da fonte é setado, por exemplo: o bit ativo tem um offset de 0 (zero) a partir do endereço principal. Entretanto, se nenhum bit está habilitado dentro da área de origem, 0 (zero) será escrito no endereço de destino e um erro é gerado.

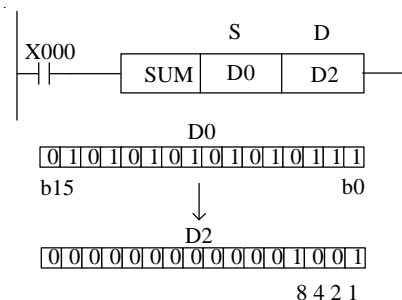
c) Quando o endereço de origem são dados ou um endereço de word, o n deve ser tirado da faixa 1 à 4 pois só existem 16 bits de origem disponíveis num única word de dados.

3.5.4 SUM (FNC 43)

Mnemônico	Função	Operandos		Passos do programa
		S	D	
SUM FNC 43 (Somatório dos bits ativos)	O número (quantidade) de bits ativos nos endereços de origem serão armazenados no endereço de destino	K, H, KnX, KnY, KnM, KnS, T, C, D, V, Z	KnY, KnM, KnS, T, C, D, V, Z	SUM,SUMP: 7 passos DSUM,DSUMP: 9 passos

Operação:

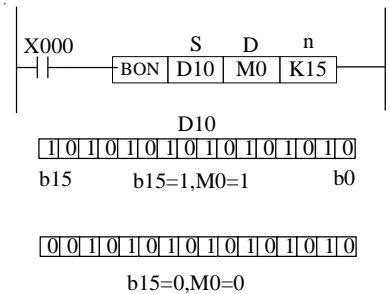
O número de bits ativos (ON) no endereço de origem (S), por exemplo: bits que possuem o valor de "1" são contados. A contagem é armazenada no registrador de destino (D). Se um formato de "double word" é usado, o endereço s de origem e destino usam registradores de 32 bits. O endereço de destino terá sempre seus 16 bits mais significativos em 0 (zero) uma vez que o valor contado nunca pode ser mais do que 32.

Se nenhum bit está habilitado então o marcador zero, M8020 é setado.

3.5.5 BON (FNC 44)

Mnemônico	Função	Operandos			Passos do programa
		S	D	n	
BON FNC 44 (Checa o status de um bit específico)	O status do bit especificado no endereço de origem será indicado no destino	K, H, KnX, KnY, KnM, KnS, T, C, D, V, Z	Y, M, S	K,H, ☆Nota: Operação de16 bits n=0 a 15 Operação de 32 bits n=0 a 31	BON, BONP: 7 passos DBONP, DBON: 13 passos

Operação:

Uma posição de um único bit (n) é especificada de dentro da área de memória de origem (S). O n poderia ser visto como um offset específico do endereço de origem (S), por exemplo: 0 (zero) sendo o primeiro endereço (um offset 0) onde como o offset de 15 seria, na verdade, o 16° endereço. Se o bit identificado torna-se ativo, por exemplo: ON, o endereço de destino (D) é ativado para "marcar" um novo status.

Poderia se dizer que o endereço de destino age como espelho do status da fonte de bits selecionada.

3.5.6 MEAN (FNC 45)

Mnemônico	Função	Operandos			Passos do programa
		S	D	n	
MEAN FNC 45 (Média)	Calcula a média de uma determinada faixa de endereços	KnX, KnY, KnM, KnS, T, C, D	KnY, KnM, KnS, T, C, D, V, Z	K,H, ☆Nota: n=1 a 64	MEAN,MEANP: 7 passos DMEAN,DMEANP: 13 passos

Operação:

A faixa de dados de origem é definida pelos operandos S e N.

S é o endereço inicial dos dados de origem e n especifica o número de endereços consecutivos que serão considerados usados.

O valor de todos os endereços dentro da faixa de origem é somado e depois dividido pelo número de endereços somados, por exemplo: n. Isto gera um valor de média inteiro que é armazenado no endereço de destino (D). O resto da divisão é ignorado.

Regra Geral

$$D = \frac{\sum_{S0}^{Sn} S}{n} = \frac{(S0+S1+...SN)}{n}$$

Exemplo

$$D10 = \frac{(D0+D1+D3)}{3}$$

Pontos a observar:

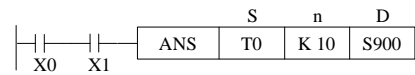
Se a área de origem especificada é verdadeiramente menor que a área fisicamente disponível, então somente os endereços disponíveis são usados. O verdadeiro valor de n usado para calcular a média refletirá os endereços disponíveis usados. Entretanto, o valor de n que foi digitado na instrução ainda será mostrado. Isto pode causar confusão uma vez que o valor da média calculado manualmente usando este valor original de n será diferente daquele que estará sendo mostrado.

Se o valor de n está especificado fora da faixa mencionada (1 a 64) um erro é gerado.

3.5.7 ANS (FNC 46)

Mnemônico	Função	Operandos			Passos do programa
		S	D	n	
ANS FNC 46 (Habilitada marcador de diagnóstico temporizado)	Esta instrução inicia um temporizador. Uma vez que a temporização é finalizada, um marcador de diagnóstico é setado	T Nota: Faixa disponível T0 à T199	S Nota: Faixa do marcador de diagnóstico S900 à S999	K,H, ☆Nota: Faixa n 1 a 32,767 – em unidades de 100mseg	ANS: 7 passos

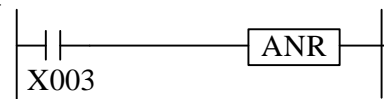
Operação:

Esta instrução, quando habilitada, inicia um temporizador (S) por um período n, com base de tempo de 100 mseg. Quando o temporizador completa sua temporização, o marcador de diagnóstico associado (D) é setado.

Se a instrução estiver desabilitada durante ou após completar o ciclo de temporização, o temporizador é automaticamente zerado. Entretanto, o status atual da bobina do marcador de diagnóstico permanecerá sem mudanças.

3.5.8 ANR (FNC 47)

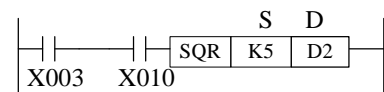
Mnemônico	Função	Operandos	Passos do programa
		D	
ANR FNC 47 (Zera marcador de diagnóstico) ★	O marcador de diagnóstico menos significativo será resetado cada vez que a instrução for executada	N/A	ANR, ANRP: 1 passo

Operação:

Os marcadores de diagnóstico que tenham sido ativados são sequencialemente zerados um-a-um, cada vez que a instrução ANR for executada. Se a instrução ANR for acionada continuamente, a mesma executará sua operação de zerar em cada varredura de programa a menos que seja modificado pelo pulso prefixo P ou por um intertravamento de programa definido por um usuário.

3.5.9 SQR (FNC 48)

Mnemônico	Função	Operandos		Passos do programa
		S	D	
SQR FNC 48 (Raiz quadrada)	Executa uma operação matemática de raiz quadrada, ex: D= $\sqrt{S}$	K,H,D	D	SQR, SQRP: 5 passos DSQR, DSQRP: 9 passos

Operação:

Esta instrução executa uma operação de raíz quadrada no dado de origem (S) e armazena o resultado no endereço de destino (D). A operação é executada com números inteiros dando a resposta arredondada. Por exemplo: se (S) = 154, então (D) é calculado como sendo 12. O M8020 é setado quando o resultado for igual a zero.

Respostas com valores arredondados ativarão o M8021.

Notas gerais:

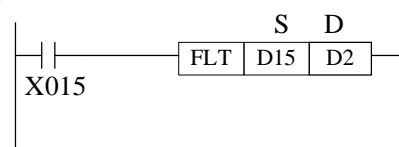
Ao executar qualquer operação de raíz quadrada (mesmo numa calculadora) e o resultado é um número negativo, isso resultará em um erro. Este erro será identificado pelo marcador especial M8067, sendo ativado:

$\sqrt{-168}$ = M8067 será setado

3.5.10 FLT (FNC 49)

Mnemônico	Função	Operandos		Passos do programa
		S	D	
FLT FNC 49 (Ponto flutuante)	Usado para converter número em ponto fixo para ponto flutuante	D		FLT, FLTP: 5 passos DFLT, DFLTP: 9 passos

Operação 1:

Quando a instrução ponto flutuante é usada, os dados de origem (S) são convertidos para um valor equivalente e armazenados no formato ponto flutuante no endereço de destino (D). Notar que dois endereços consecutivos (D e D+1) serão usados para armazenar o número convertido. É verdade que independente do tamanho dos dados de origem (S), mesmo que (S) seja uma word (16 bits) ou uma double word (32 bits), isso não afetará no número de endereços (D) usados para armazenar o número em ponto flutuante. Exemplos:

Dados da origem em decimal (S)	Valor do destino do ponto de floating (D)
1	1
-26700	-2.67×10^4
404	4.04×10^2

Pontos a observar:

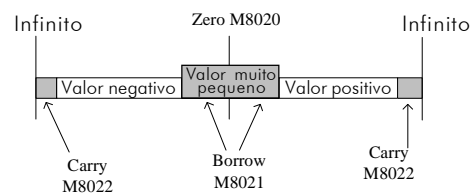
a) Quando números de ponto flutuante são usados, os marcadores de carry, borrow e zero (M8020, M8021 e M8022 respectivamente) operam da seguinte forma:

M8020, Zero: é ativado quando o resultado é Zero.

M8021, Borrow: é ativado quando o resultado é menor que o menor número possível.

O resultado é forçado a ser igual ao menor número e um marcador associado é setado.

M8022, Carry: é ativado quando o resultado é maior que o maior número possível. O resultado é forçado ser igual ao maior número e o marcador associado é setado.

b) Números em ponto flutuante sempre ocuparão 32 bits consecutivos, ex: 2 registros de dados consecutivos. Ao converter números de ponto fixo para ponto flutuante, deve-se prever endereços de destino suficientes, ex:

Instrução	Operação double word	Status do M8023	Número de registradores de origem (S)	Número dos registros de destino (S)	Comentário
FLT	NÃO	OFF	1(S)	2(D,D+1)	Conversão para ponto flutuante
FLT(INT)		ON	2(S,S+1)	1(D)	Conversão para decimal
DFLT	SIM	OFF	2(S,S+1)	2(D,D+1)	Conversão para ponto flutuante
DFLT(DINT)		ON	2(S,S+1)	2(D,D+1)	Conversão para decimal

3.6 PROCESSAMENTO DE ALTA VELOCIDADE - FUNÇÕES 50 À 59

Conteúdo:

Lista de símbolos:

D – Endereço de destino.

S – Endereço de origem.

m, n - Número de endereços ativos, bits ou uma constante operacional.

Sufixos numéricos adicionais serão anexados se houver mais de um operando com a mesma função, ex: D1, S3 ou para endereços de lista/tabelados D3+0, S+9 etc.

MSB – Bit mais significante, por vezes usado para indicar o sinal matemático de um número, ex: positivo = 0, e negativo = 1.

LSB – Bit menos significante.

Modificações de instrução:

☐☐☐- Instrução de 16 bits, onde ☐☐☐ identifica a instrução mnemônica.

☐☐☐P - Instrução de 16 bits habilitada por pulso (simples).

D☐☐☐- Instrução de 32 bits.

D☐☐☐P - Instrução de 32 bits habilitada por pulso (simples).

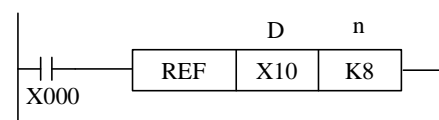
★ - Uma instrução repetitiva que mudará o valor de destino a cada varredura menos quando for habilitada por um pulso.

☆ - Um operando que não pode ser indexado, ex: A adição de V ou Z ou é inválida ou não terá efeito para o valor do operando.

3.6.1 REF (FNC 50)

Mnemônico	Função	Operandos		Passos do programa
		D	n	
REF FNC 50 (Refresh) ★	Força a atualização imediata das entradas ou saídas especificadas	X, Y , ☆ Nota: D deveria sempre ser um múltiplo de 10, ex: 00, 10, 20, 30 etc.	K, H, ☆ Nota: n deveria sempre ser um múltiplo de 8, ex: 8, 16, 24, 32 etc	REF, REFP: 5 passos

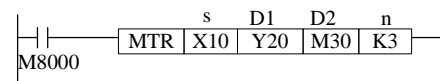
Operação:

A operação padrão do PLC, ele atualiza o estado das entradas e saída ao final de cada ciclo de varredura (instrução END). Se uma atualização imediata do status de um endereço I/O é necessária, então a instrução REF é usada. A instrução REF só pode ser usada para atualizar blocos de 8 (n) endereços consecutivos. O endereço inicial a ser atualizado deveria sempre ter seu último dígito como 0 (zero), por exemplo, em unidades de 10.

Nota: Ocorrerá um pequeno atraso antes do endereço de I/O ser fisicamente atualizado. No caso de entradas, um tempo equivalente ao definido para o filtro, enquanto as saídas terão apenas o retardo elétrico.

3.6.2 MTR (FNC 52)

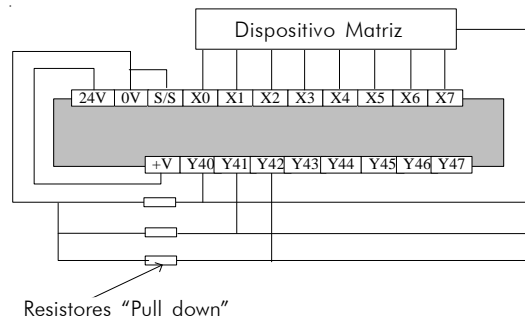
Mnemônico	Função	Operandos				Passos do programa
		S	D1	D2	n	
MTR FNC 52 (Entrada da matriz)	Armazena os dados de entrada numa matriz. Só pode ser usada UMA vez.	X ✰	Y ✰	Y,M,S ✰	K,H ✰	MTR: 9 passos
		Nota: Estes operandos deveriam ser sempre um múltiplo de 10, ex: 00, 10, 20, 30 etc.			Nota: n=2 a 8	

Operação:

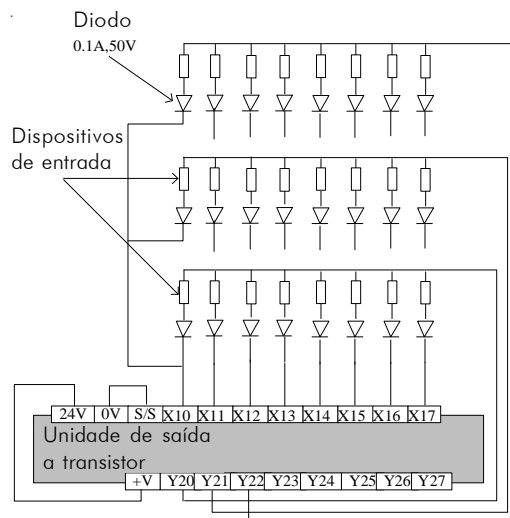
Esta instrução permite que uma seleção de 8 endereços consecutivos (endereço inicial S) sejam usados múltiplas (n) vezes, por exemplo, cada entrada física tem mais de um sinal separado e bastante diferente (D1) sendo processado. O resultado é armazenado numa tabela-matriz (endereço inicial D2).

Pontos a observar:

a) A instrução MTR envolve o chaveamento de entradas e saídas de alta velocidade. Por esta razão esta instrução só é recomendada para uso com módulos de saída de transistor.
b) Para a instrução MTR operar corretamente, ela deve ser acionada continuamente. Recomenda-se que o marcado auxiliar M8000, o marcador de status RUN do PLC seja usado. Após completar a primeira leitura completa da matriz, o marcador de operação completa M8029 é setado. Este marcador é automaticamente zerado quando a instrução MTR é desabilitada.
c) Cada conjunto de 8 sinais de entrada é agrupado num ‘banco’ (há um número n de bancos).
d) Cada banco é habilitado por uma saída dedicada (endereço inicial D1). Isto significa que a quantidade de saídas de D1 usadas para alcançar a matriz é igual ao número de bancos n.
Como agora há entradas adicionais entrando o PLC, cada uma terá um status que precisa ser registrado. Isto é armazenado na tabela-matriz. Esta tabela-matriz inicia no endereço principal D2. A construção da matriz imita o mesmo sinal 8 por configuração de banco n. Portanto, quando uma certa entrada num banco selecionado é lida, seu status é armazenado numa posição equivalente no resultado da tabela-matriz.
e) A instrução matriz opera num formato de interrupção, processando cada banco de entradas a cada 20mseg. Este tempo é baseado nos filtros de entrada selecionados na programados em 10mseg. Isto resultaria numa matriz 8x8, por exemplo, 64 entradas (8 entradas registradas em 8 bancos) lidas em160mseg.

Se entradas de alta velocidade (ex. X0) são especificadas para o operando S, o tempo de leitura de cada área se torna somente 10mseg, ou seja, a metade da velocidade de leitura. Entretanto, resistores "pull down" adicionais são requisitados nas saídas acionadas para garantir que a leitura de alta velocidade não detecte quaisquer correntes residuais da última operação.

Estas deveriam ser colocadas em paralelo ao sinal de entrada e deveriam ter o valor de aproximadamente 3.3KΩ, 0.5W. Para uso mais fácil, entradas de alta velocidade não deveriam ser especificadas em S.

f) Porque esta instrução usa uma série de sinais multiplexados, ela requer certa quantidade de fiação para operar. O diagrama de interligação acima da um exemplo de um circuito usado se a instrução do exemplo anterior foi programada. Como precaução geral para auxiliar numa operação bem sucedida, diodos deveriam ser colocados após cada dispositivo de entrada (ver diagrama ao lado). Estes deveriam ser na faixa de 0.1A, 50V.

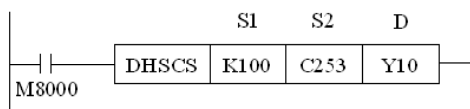
g) Exemplo de Operação

Quando a saída Y20 está energizada somente as entradas do primeiro grupo são lidas. Estes resultados são então armazenados; neste exemplo, nos marcadores auxiliares M30 à M37. O segundo passo envolve Y20 desenergizada e Y21 ligada. Desta vez somente entradas no segundo grupo são lidas. Estes resultados são armazenados nos endereços M40 a M47. O último passo deste exemplo tem Y21 desligada e Y22 ligada. Isto então permite que todas as entradas no terceiro grupo sejam lidas e armazenadas nos endereços M50 à M57. O processamento deste exemplo de instrução levaria 20X3 = 60mseg.

3.6.3 HSCS (FNC 53)

Mnemônico	Função	Operandos			Passos do programa
		S1	S2	D	
HSCS FNC 53 (Habilita o contador de alta velocidade)	Habilita a saída selecionada quando o valor do contador de alta velocidade especificado é igual ao valor do teste	K, H, KnX, KnY, KnM, KnS, T, C, D, Z	C Nota: C = 235 a 249, 251 a 254	Y, M, S Ponteiros de interrupção I010 à I060 podem ser definidos	DHSCS: 13 passos

Operação:

O conjunto HSCS, compara o valor atual do contador de alta velocidade selecionado (S2) à um valor selecionado (S1). Quando o valor atual dos contadores muda para um valor igual a S1, o endereço de destino (D) é setado. O exemplo acima mostra que

Y10 estaria energizada somente quando o valor de C253 fosse de 99-100 OU 101-100. Se o valor atual dos contadores fosse forçado a ser igual a 100, a saída Y10 NÃO estaria energizada.

Pontos a observar:

a) Recomenda-se que a entrada de habilitação usada para as funções do contador de alta velocidade, HSCS,HSCR, HSCZ seja o marcador auxiliar especial RUN M8000.

b) Se mais de uma função do contador de alta velocidade é usada para um único contador, os endereços (D) deveriam ser mantidos dentro de 1 grupo de 8 endereços, ex: Y0-7, M10-17.

c) Todas as funções do contador de alta velocidade usam um processo de interrupção, portanto, todos os endereços destino (D) são imediatamente atualizados.

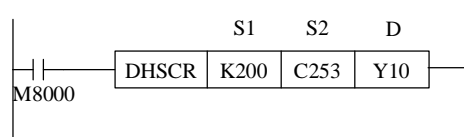
Use dos ponteiros de interrupção

Podem-se usar ponteiros de interrupção de I010 à I060 (6 pontos) como endereços de destino (D). Isto habilita imediatamente as rotinas de interrupção quando o valor do contador de alta velocidade especificado alcança o valor na instrução HSCS.

3.6.4 HSCR (FNC 54)

Mnemônico	Função	Operandos			Passos do programa
		S1	S2	D	
HSCR FNC 54 (Zera contador de alta velocidade)	Zera a saída selecionada quando o contador de alta velocidade especificada é igual ao valor do teste	K, H, KnX, KnY, KnM, KnS, T, C, D, Z	C Nota: C = C235 a C249,C251 a C254	Y, M, S,C Nota: Se C, use o mesmo contador que S1	DHSCR: 13 passos

Operação:

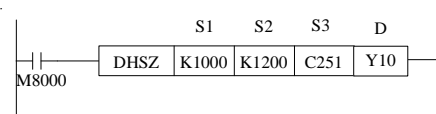
O HSCR compara o valor atual do contador de alta velocidade selecionado (S2) à um valor selecionado (S1). Quando o valor atual dos contadores muda para um valor igual a S1, o endereço de destino (D) é zerado. No exemplo acima, Y10 seria zerado somente quando o valor de C253 fosse de 199-200 ou 201-200. Se o valor atual de C253 fosse forçado a ser igual a 200, a saída Y10 NÃO seria zerada.

Para outros pontos gerais sobre o uso de funções de contadores de alta velocidade, ver a subseção 'Pontos a observar' em HSCS (FNC 53). Pontos relevantes são: a, b e c.

Fazer referência à nota sobre o número de instruções de alta velocidade permitidas.

3.6.5 HSZ (FNC 55)

Mnemônico	Função	Operandos				Passos do programa
		S1	S2	S3	n	
HSZ FNC 55 (Compara a zona de alta velocidade)	Operação 1: O valor atual de um contador de alta velocidade é verificado numa faixa especificada	K, H, KnX, KnY, KnM, KnS, T, C, D, Z S1≤ S2		C Nota: C = 235 a 249 , C251 a C253	Y, M, S Nota: 3 endereços consecutivos são usados	DHSZ: 17 passos

Operação 1 - Padrão:

Esta instrução funciona exatamente da mesma forma que o Padrão ZCP (FNC11). A única diferença é que o endereço que está sendo comparado é um contador alta velocidade (especificado como S3).

Também, todas as saídas (D) são atualizadas imediatamente devido à interrupção da operação do DHSZ. Deve-se lembrar que quando um endereço é especificado no operando D, ele é de fato o primeiro endereço de 3 endereços consecutivos. Cada um é usado para representar o status da comparação atual, por exemplo, usando o exemplo acima como base:

Y10 (D) C251 é menor que S1, K1000 (S3< S1)

Y11 (D+1)C251 é maior que ou igual a S1, K1000, mas menor que ou igual a S2, K1200 (S3≥ S1, S3≤ S2)

Y12 (D+2)C251 é maior que S2, K1200 (S3> S2)

Para outros pontos gerais sobre o uso de funções de contadores de alta velocidade, ver a subseção 'Pontos a observar' em HSCS (FNC 53). Pontos relevantes são: a, b e c.

Fazer referência à nota sobre o número de instruções de alta velocidade permitidas.

3.6.6 SPD (FNC 56)

Mnemônico	Função	Operandos			Passos do programa
		S1	S2	D	
SPD FNC 56 (Detecção da velocidade)	Detecta o número de pulsos enviados num dado período de tempo. Resultados podem ser usados para calcular a velocidade	X0 à X5	K, H, KnX, KnY, KnM, KnS, T, C, D, V, Z Unidade é mseg	T, C, D, Z (V) Nota: 3 endereços consecutivos são usados. No caso de D= Z monitor D8028, D8029 e D8030	SPD: 7 passos

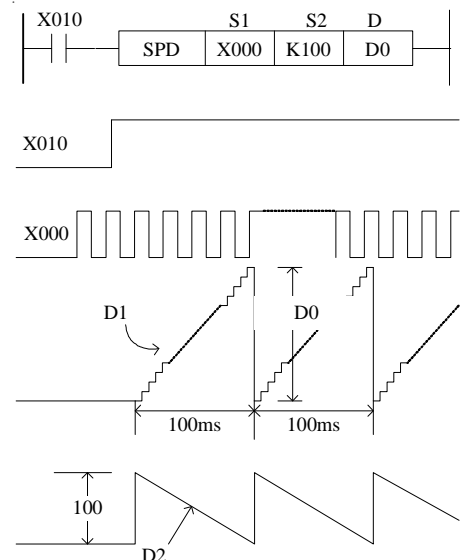
Operação:

O número de pulsos recebidos em S1 são contados e armazenados em D+1, este é o valor de contagem atual.

A contagem acontece dentro de um determinado período de tempo especificado por S2 em mseg. O tempo restante na atual contagem é mostrado no dispositivo D+2.

Os números de pulsos contados (do S1) da última contagem temporizada são armazenados em D. A tabela de tempos ao lado mostra a operação SPD em gráfico.

Nota: O valor da contagem atual, endereço D+1. valor acumulado/ última contagem, endereço D.

Tempo restante atual em mseg, dispositivo D+2.

Pontos a observar:

a) Quando o período de tempo é finalizado, os dados armazenados em D+1 são imediatamente escritos em D. O D+1 é então zerado e um novo período de tempo é iniciado.

b) Por se tratar de um processo de alta velocidade e interrupção, somente entradas X0 à X5 podem ser usadas como endereço de origem S1. Entretanto, o endereço especificado para S1 NÃO deve coincidir com qualquer outra função de alta velocidade que esteja operando, por exemplo, um contador de alta velocidade usando a mesma entrada. A instrução SPD age como um contador monofásico.

c) Múltiplas instruções de SPD podem ser usadas, mas os endereços de origem S1 identificados restringem isto a um máximo de 6 vezes.

d) Uma vez que os valores para os pulsos contados forem coletados, velocidades apropriadas podem ser calculadas usando matemática simples. Estas velocidades poderiam ser velocidades radiais em RPM, velocidades lineares em M/min. Depende totalmente da manipulação matemática colocada nos resultados SPD. As seguintes interpretações podem ser usadas:

$$\text{Velocidade linear (km/h)} = \frac{3600 \times (D)}{n \times S2} \times 10^3$$

Onde n = o número de divisões do encoder linear por quilometro.

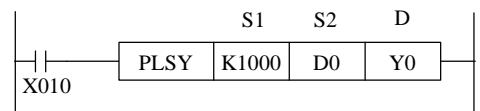
$$\text{Velocidade radial (rpm)} = \frac{60 \times (D)}{n \times S2} \times 10^3$$

Onde n = o número de pulsos por volta do encoder.

3.6.7 PLSY (FNC 57)

Mnemônico	Função	Operandos			Passos do programa
		S1	S2	D	
PLSY FNC 57 (Saída de pulso Y)	Gera um determinado número de pulso numa freqüência definida numa saída Y	K, H, KnX, KnY, KnM, KnS, T, C, D, V, Z		Y: Somente Y000 e Y001	PLSY: 7 passos DPLSY: 13passos

Operação:

Uma quantidade especifica de pulsos S2 é gerada através da saída do endereço D numa freqüência especifica S1. Esta instrução é usada em situações onde o número de pulsos gerados numa determinada saída é vital para o processo.

Pontos a observar:

a)A freqüência máxima:

16 bits: 1~32767 Hz

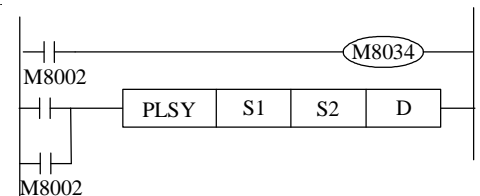
32 bits: 1~100000 HZ

b) O número máximo de pulsos: Operação de 16 bits: 1 à 32.767 pulsos, Operação de 32 bits: 1 à 2.147.483.647 pulsos.

Nota: O marcador auxiliar M8029 é setado quando o número de pulsos especificado foi gerado. A contagem de pulsos e o marcador de processo finalizado (M8029) são zerados quando a instrução PLSY é desabilitada. Se "0" (zero) é especificado a instrução PLSY continuará a gerar pulsos enquanto a instrução estiver habilitada.

c) Um único pulso é descrito como tendo uma duração de 50% do ciclo completo de geração, isto significa que durante 50% do pulso o sinal de saída estará energizado e, conseqüentemente os 50% restantes do pulso, a saída estará desenergizada. A saída, na verdade, é controlada por interrupção, ou seja, a freqüência de chaveamento da saída NÃO é afetada pelo tempo de varredura do programa.

d) Os dados nos operandos S1 e S2 podem ser trocados durantes a execução. Entretanto os novos dados em S2 não se tornarão efetivos até que a operação atual tenha sido completada, ou seja, a instrução tem que ser desabilitada retirando-se o contato de habilitação da mesma.

e) Esta instrução só pode ser usada uma vez numa varredura de programa. Também, somente FNC57 PLSY ou FNC 59 PLSR podem estar ativas no programa. É possível usar sub-rotinas ou outras técnicas de programação parecidas para evitar que estas instruções sejam habilitadas ao mesmo tempo. Neste caso, a instrução atual deve ser desativada antes de habilitar a próxima.

f) Dependendo da freqüência de atuação da saída, recomenda-se o uso de unidades de saída a transistor. Para frequências altas, saídas a relé terão uma vida extremamente reduzida e farão com que sinais de saída indesejáveis ocorram devido ao repique mecânico dos contatos. Para garantir um sinal de saída 'limpo' ao usar unidades a transistor, a corrente da carga deveria ser de 200mA ou mais. Pode ser que resistores 'pull up' sejam necessários.

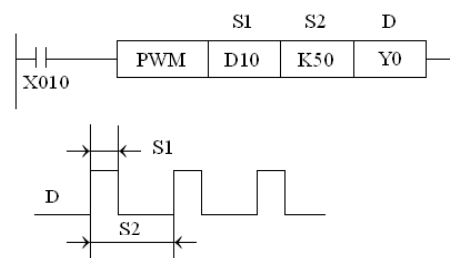
3.6.8 PWM (FNC 58)

Mnemônico	Função	Operandos			Passos do programa
		S1	S2	D	
PWM FNC 58 (Saída PWM)	Gera um trem de pulso com características de pulso definidas	K, H, KnX, KnY, KnM, KnS, T, C, D, V, Z Nota: S1≤ S2		Y: Somente Y000 e Y001	PWM: 7 passos

Operação:

Um trem de pulso contínuo é gerado através do endereço D quando esta instrução é acionada. As características do pulso são definidas como:

O período do ciclo do pulso, em tempo (mseg), entre duas partes idênticas de pulsos consecutivos (S2).

E também por quanto tempo (mseg), o nível alto do pulso deverá existir (S1).

Pontos a observar:

a) Esta é uma instrução de16 bits, as faixas de tempo disponíveis para S1 e S2 são 1 a 3000.

b) O cálculo do período do pulso é facilmente feito dividindo S1 por S2. Portanto, S1 não pode ter um valor maior que S2, isto significaria que o pulso estaria ligado por mais tempo que o ciclo total do pulso, ou seja, um segundo pulso iniciaria antes do primeiro terminar. Se isto é programado um erro ocorrerá.
Esta instrução é usada onde se deseja controlar o comprimento do pulso.

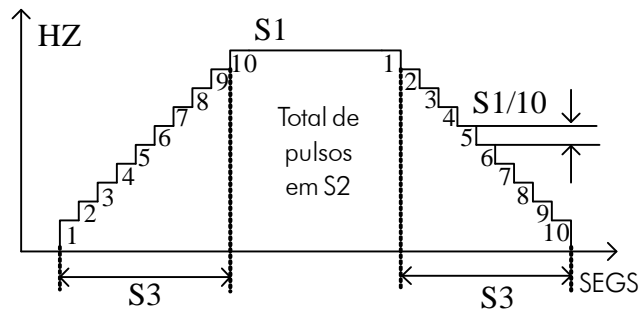
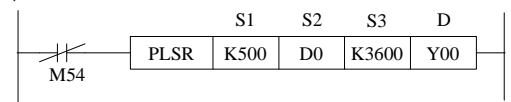
c) A instrução PWM só poderá ser usada uma vez num programa de usuário.

d) Dependendo da freqüência de atuação da saída, recomenda-se o uso de unidades de saída a transistor. Para frequências altas, saídas a relé terão uma vida extremamente reduzida e farão com que sinais de saída indesejáveis ocorram devido ao repique mecânico dos contatos. Para garantir um sinal de saída 'limpo' ao usar unidades a transistor, a corrente da carga deveria ser de 200mA ou mais. Pode ser que resistores 'pull up' sejam necessários.

3.6.9 PLSR (FNC 59)

Mnemônico	Função	Operandos				Passos do programa
		S1	S2	S3	D	
PLSR FNC 59 (Rampa)	Gera uma rampa de aceleração e desaceleração num determinado período de tempo	K, H,KnX, KnY, KnM, KnS, T, C, D, V, Z			Y: Somente Y000 e Y001	PLSR: 9 passos DPLSR: 17 passos

Operação:

Uma quantidade especificada de pulsos S2 é gerada através do endereço D. A freqüência de saída é elevada em rampa em 10 passos até a freqüência máxima S1 num determinado tempo de aceleração S3 ms, então é reduzida à zero até parar também em S3 ms. Esta instrução é usada para gerar curvas de aceleração/desaceleração simples onde a deseja-se controlador o tempo da rampa de aceleração e desaceleração.

Pontos a observar:

☑ Usuários podem usar freqüências de 10 a 100.000Hz. A freqüência deve ser programada em múltiplos de 10. Caso contrário, o valor será arredondado para o próximo múltiplo de 10.
Os passos de aceleração e desaceleração são programados para 1/10 da freqüência máxima. Levar isto em consideração para evitar escorregamento ao usar motores de passo.

☑ O número máximo de pulsos: Operação de 16 bits: 110 à 32.767 pulsos, Operação de 32 bits: 110 à 2.147.483.647 pulsos.
Saída de pulso correta não pode ser garantida para uma programação de 110.

☑ O tempo de aceleração deve estar em conformidade com as limitações descritas na página a seguir.

☑ O endereço de saída está limitado a Y0 ou Y1 somente e deve ser do tipo transistor.

i) Esta instrução só pode ser usada uma vez numa varredura de programa. Também, somente um FNC 57 PLSY ou FNC 59 PLSR pode ser ativo no programa de uma vez. É possível usar sub-rotinas ou outras técnicas de programação para evitar que estas instruções sejam habilitadas simultaneamente. A instrução atual deve ser desativada antes de habilitar a próxima.
Se o número de pulsos não é o suficiente para alcançar a freqüência máxima então a freqüência é automaticamente cortada.
O marcador auxiliar M8029 é setado quando o número de pulsos especificado for atingido. A contagem de pulsos e o marcador (M8029) são zerados quando a instrução PLSR é desabilitada.

Limitações do tempo de aceleração

O tempo de aceleração S3 tem como limite máximo de 5000 ms. Entretanto, os verdadeiros limites do S3 são determinados por outros parâmetros do sistema de acordo com os 4 pontos a seguir:

1) Programar o S3 para ser mais do que 10 vezes o tempo máximo de varredura de programa (D8012).

Se programado para menos do que isso, então a temporização dos passos de aceleração não serão coerentes.

1) A fórmula a seguir fornece o valor mínimo de S3.

$$S3 \geqslant \frac{9000}{S1} \times 5$$

2) A fórmula a seguir fornece o valor máximo de S3.

$$S3 \leqslant \frac{S2}{S1} \times 818$$

4) A saída de pulso sempre incrementa em 10 passos até a máxima freqüência como mostrado na página anterior.

Se os parâmetros não vão de encontro às condições acima, reduza o tamanho de S1.

☑ A freqüência de saída possível está limitada em 10 à 100,000 Hz. Se a freqüência máxima ou o tamanho do passo de aceleração estão fora deste limite então eles são automaticamente ajustados para trazer o valor de volta para o limite.

☑ Se o sinal de habilitação está desligado, todas as saídas param. Quando a habilitação é setada novamente, o processo inicia do começo.

☑ Mesmo se os operandos são trocados durante a operação, o perfil da saída não muda. Os novos valores tornam-se efetivos na próxima operação.

3.7 INSTRUÇÕES ÚTEIS – FUNÇÕES 60 À 69

Conteúdo:

IST -	Estado inicial	FNC 60
SER -	Procurar dado numa pilha	FNC 61
ABSD -	Seqüenciador absoluto	FNC 62
INCD -	Seqüenciador incremental	FNC 63
TTMR -	Monitoração de tempo	FNC 64
STMR -	Temporizador especial	FNC 65
ALT -	Inversão de estado	FNC 66
RAMP -	Rampa para variação de valor	FNC 67
ROTC -	Monitor de rotação	FNC 68
SORT -	Seleção de dados numa tabela	FNC 69

Lista de símbolos:

D – Endereço de destino.

S – Endereço de origem.

m, n - Número de endereços ativos, bits ou uma constante operacional.

Sufixos numéricos adicionais serão anexados se houver mais de um operando com a mesma função, ex: D1, S3 ou para endereços de lista/tabelados D3+0, S+9 etc.

MSB – Bit mais significante, por vezes usado para indicar o sinal matemático de um número, ex: positivo = 0, e negativo = 1.

LSB – Bit menos significante.

Modificações de instrução:

☐☐☐ - Instrução de 16 bits, onde ☐☐☐ identifica a instrução mnemônica.

☐☐☐P - Instrução de 16 bits habilitada por pulso (simples).

D☐☐☐ - Instrução de 32 bits.

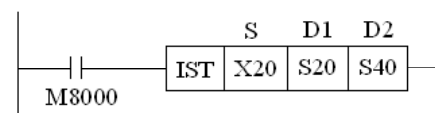
D☐☐☐P - Instrução de 32 bits habilitada por pulso (simples).

★ - Uma instrução repetitiva que mudará o valor de destino a cada varredura menos quando for habilitada por um pulso.

☆ - Um operando que não pode ser indexado, ex: A adição de V ou Z ou é inválida ou não terá efeito para o valor do operando.

3.7.1 IST (FNC 60)

Mnemônico	Função	Operandos			Passos do programa
		S	D1	D2	
IST FNC 60 (Estado inicial)	Automaticamente define um sistema de operação STL multi-modo	X, Y, M, S, Nota: Usa 8 endereços consecutivos	S, Nota: S20~S1023，D1 deve ser menor que D2		IST: 7 passos

a) Esta instrução IST automaticamente designa e usa diversos marcadores de bits e word. Estes são listados na coluna no lado direito desta página.

b) A instrução IST só pode ser usada UMA vez.
A instrução deve ser programada o mais próximo do início do programa.

c) O modo de operação requerida é selecionado acionando os endereços associados com operandos S+0 até S+4(5 entradas). Nenhum dos endereços nesta faixa deve ser setado ao mesmo tempo. Recomenda-se que estas 'entradas' sejam selecionadas através do uso de uma chave de seleção.
Se o modo de operação selecionado é mudado antes do marcador 'retorno zero completo' (M8043) ser programado, todas as saídas serão desenergizadas.

d) 'Posição zero' é um termo usado para identificar uma posição de dados onde o endereço controlado começa e retorna após completar sua tarefa. Portanto, o modo de operação 'retorno zero' faz com que o sistema controlado retorne para estes dados.

e) Os modos de operação disponíveis são divididos em dois grupos principais, manual e automático. Há sub-modos para estes grupos.

Sua operação é definida como:

Manual

Manual (selecionado pelo endereço S+0) – É possível acionar cargas individuais de acordo com um comando específico, por exemplo, o uso de botões.

Retorno Zero (selecionado pelo dispositivo S+1) – As saídas são devolvidas aos seus estados iniciais quando a entrada Zero (S+5) é dada.

Automático

Um passo (selecionado pelo endereço S+2) - A seqüência controlada opera automaticamente, mas só segue para o próximo passo quando a entrada de habilitação (S+6) é dada.

Um ciclo (selecionado pelo endereço S+3) – As saídas serão acionadas por um ciclo de operação. Depois que o ciclo foi completado, as saídas são acionadas conforme seu estado inicial, na posição 'zero'. O ciclo é iniciado depois que uma entrada de início (S+6) é dada.

Um ciclo que está na verdade sendo processado pode ser parado a qualquer momento ativando a entrada de 'parada' (S+7). Para reiniciar a seqüência da posição 'pausada' a entrada de início deve ser dada mais uma vez.

Automático (selecionado pelo endereço S+4) - Uma operação totalmente automática é possível neste modo. O ciclo programado é executado repetidamente quando a entrada de 'início' (S+6) é dada. O ciclo de operação não parará imediatamente quando a entrada de 'parar' (S+7) é dada.

A operação atual continuará para terminar o ciclo atual e então para sua operação.

Nota: Entradas de início, parar e zero são freqüentemente dadas por sinais externos operados manualmente.

Notar que a entrada 'parar' é somente um sinal de parar de programa. Não pode ser usado como substituto de um botão de 'Parada de emergência'. Todos os endereços de segurança, 'Parada de emergência' e etc., deveriam ser sistemas de atuação física que efetivamente isolarão a máquina da operação e cortaram a alimentação externa da mesma. Referir-se a padrões locais e nacionais para práticas de segurança aplicáveis.

Dispositivos designados

Endereços selecionados pelo usuário indireto:
S+0 Operação manual
S+1 Retorno zero
S+2 Operação de passo
S+3 Operação de um ciclo
S+4 Operação cíclica
S+5 Início retorno zero
S+6 Início de operação automática
S+7 Parar

Estados iniciais:
S0 inicia operação ‘manual’
S1 inicia operação de ‘retorno zero’
S2 inicia operação ‘automática’

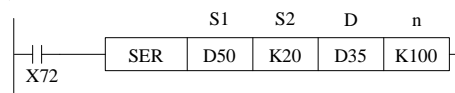
Estados gerais:
Seqüência ‘retorno zero’ S10 a S19
Seqüência ‘retorno automático’ D1 a D2

Marcadores de bits especiais:
M8040 = ON - transferência de estado STL é inibido
M8041 = ON - estados iniciais são habilitados
M8042 = Pulso inicial dado pela entrada de “start”
M8043 = ON - retorno zero completado
M8044 = ON - zero máquina detectado
M8047 = ON - monitor STL habilitado

3.7.2 SER (FNC 61)

Mnemônico	Função	Operandos				Passos do programa
		S1	S2	D	n	
SER FNC 61 (Procurar um dado numa pilha)	Gera uma lista de estatísticas sobre um único valor de dados localizado em uma pilha de dados	KnX, KnY, KnM, KnS, T, C, D	KnX, KnY, KnM, KnS, T, C, D,V, Z K, H	KnY, KnM, KnS T, C, D Nota: 5 endereços consecutivos são usados	K,H, D ☆ Nota: n= 1~256 para operação de 16 bits n= 1~128 para operação de 32 bits	SER, SERP: 9 passos DSER, DSERP: 17 passos

Operação:

A instrução SER procura por um dado numa pilha definida pelo endereço inicial S1, com um comprimento de n dados. Os dados procurados são especificados no parâmetro S2 e os resultados da busca são armazenados no endereço de destino D por 5 endereços consecutivos.

Dispositivo de destino	Descrição do dispositivo
D	Número total de ocorrências do valor procurado S2 (0 se nenhuma ocorrência é encontrada)
D+1	A posição (na pilha do dado procurado) da primeira ocorrência do valor procurado S2
D+2	A posição (na pilha do dado procurado) da última ocorrência do valor procurado S2
D+3	A posição (na pilha do dado procurado) do menor valor encontrado na pilha de dados (a última ocorrência é devolvida se há múltiplas ocorrências com o mesmo valor)
D+4	A posição (na pilha do dado procurado) do maior valor encontrado na pilha de dados (a última ocorrência é devolvida se há múltiplas ocorrências com o mesmo valor)

Os dados procurados são especificados no parâmetro S2 e os resultados da busca são armazenados no endereço de destino D por 5 endereços consecutivos.

Pontos a observar:

a) Regras normais de álgebra são usadas para determinar os maiores e menores valores, ex: -30 é menor que 6 etc.

b) Se nenhuma ocorrência dos dados procurados é encontrada então os endereços de destino D, D+1 e D+2 serão iguais a 0 (zero).

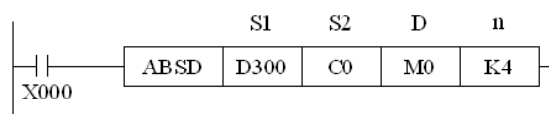
c) Ao usar o registro de dados como o endereço de destino D, lembrar que a operação de 16 bits ocupará 5 registros de dados consecutivos, mas uma operação de 32 bits ocupará 10 registros de dados em pares formando 5 words duplas.

d) Quando endereços de bits múltiplos são usados para armazenar o resultado (independente de ser uma operação de 16 ou 32 bits), somente o tamanho especificado do grupo está escrito para 5 ocorrências consecutivas, ex: K1Y0 ocuparia 20 endereços de bits de Y0 (K1 = 4 dispositivos de bits e haverá 5 grupos para os 5 resultados). Como a pilha de dados tem no máximo 256 (0 à 255) entradas de comprimento, o grupo otimizado de endereços de bits requerido é K2, ex: endereços de 8 bits.

3.7.3 ABSD (FNC 62)

Mnemônico	Função	Operandos				Passos do programa
		S1	S2	D	n	
ABSD FNC 62 (Seqüenciador absoluto)	Gera padrões de saída múltipla em resposta a dados do contador	KnX, KnY,KnM, KnS, (16 bits, n=4;32 bits, n=8),T, C, D	C 16 bits, C0~C199; 32 bits, C200~C255	Y,M,S	K,H ☆ Nota: N≤64	ABSD: 9 passos DABSD: 17 passos.

Operação:

Esta instrução gera uma variedade de padrões de saída (há um número n de saídas endereçadas) em resposta ao valor atual de um contador selecionado, S2.

Pontos a observar:

a) O valor atual do contador selecionado (S2) é comparado com uma tabela de dados definidos pelo usuário. Esta tabela de dados tem um endereço inicial identificado pelo operando S1. O S1 deveria sempre ter um número de endereço par.

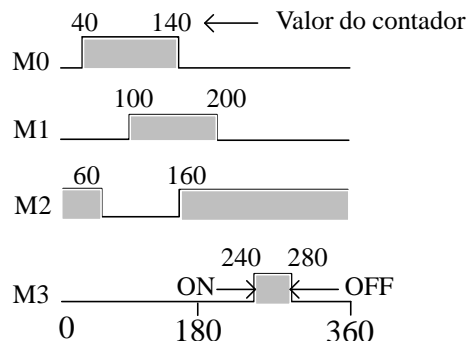
b) Para cada bit de destino (D) há dois valores consecutivos armazenados na tabela de dados. O primeiro valor alocado representa o número do evento quando o endereço de destino (D) é setado. O segundo identifica o evento de zerar. Os valores da tabela de dados são alocados como um par consecutivo para cada elemento seqüencial entre D e D+n.

c) A tabela de dados tem um comprimento igual a 2 x n entradas de dados. Dependendo do formato da tabela de dados, uma simples entrada pode ser uma word de dados como D300 ou um grupo de endereço de 16 bits, ex: K4X000.

d) Valores de 0 a 32.767 podem ser usados na tabela de dados.

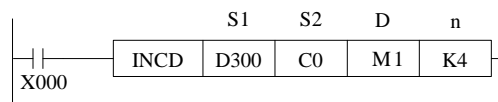
e) A instrução ABSD só pode ser usada UMA vez.

Do exemplo de instrução e a tabela de dados abaixo, o seguinte diagrama de tempos para os elementos M0 a M3 podem ser construídos.

Quando o contador S2 é igual ao valor abaixo, o endereço de destino D é setado		Endereço de destino D definido
ON	OFF	
D300 - 40	D301 - 140	M0
D302 - 100	D303 - 200	M1
D304 - 160	D305 - 60	M2
D306 - 240	D307 - 280	M3

3.7.4 INCD (FNC 63)

Mnemônico	Função	Operandos S1	Operandos S2	Operandos D	Operandos n	Passos do programa
INCD FNC 63 (Seqüenciador incremental)	Gera uma única seqüência de saída em resposta aos dados do contador	KnX, KnY, KnM, KnS, (16 bits, n=4) T, C, D	C Usa 2 Contadores consecutivos C0~C198	Y, M, S	K,H ✰ Nota: N≤64	INCD: 9 passos

Operação:

Esta instrução gera uma seqüência de padrões de saída seqüencial (há um número n de saídas endereçadas) em reposta ao valor atual de um par de contadores selecionados (S2, S2+1).

Pontos a observar:

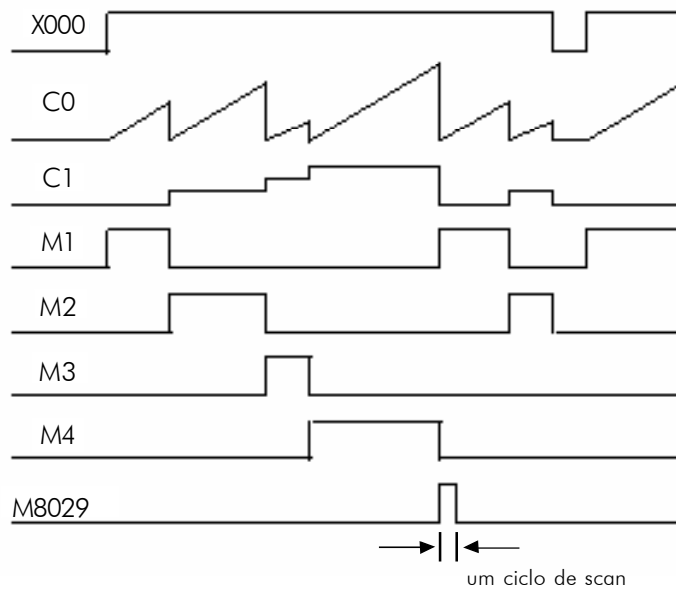
a) Esta instrução usa uma 'tabela de dados' que contem uma única lista de valores que deverão ser selecionados e comparados por dois contadores consecutivos (S2 e S2+1). A tabela de dados é identificada como tendo um endereço inicial S1 e consiste de n elementos de dados.

b) O contador S2 está programado de forma convencional. O valor programado para o contador S2 DEVE ser maior que quaisquer dos outros valores inseridos na tabela de dados. O contador S2 conta um evento de usuário e compara-o com o valor dos elementos de dados selecionados da tabela de dados.
Quando o contador e os valores de dados são iguais, o S2 incrementa a contagem do contador S2+1 e zera seu próprio valor atual para '0' (zero). Este novo valor do contador S2+1 seleciona os novos elementos de dados da tabela de dados e o contador S2 agora compara com os novos valores dos elementos de dados.

c) O contador S2+1 pode ter valores de 0 a n. Uma vez que o elemento de dados na posição n é processado, o marcador M8029 é setado. Isto então automaticamente zera o contador S2+1, portanto, o ciclo inicia novamente com o elemento de dados S1+0.

d) Valores de 0 à 32.767 podem ser usados na tabela de dados.

e) A instrução INCD só pode ser usada UMA vez. Do exemplo de instrução e da tabela de identificada abaixo, o diagrama de tempos a seguir para os elementos M1 a M4 pode ser construído.

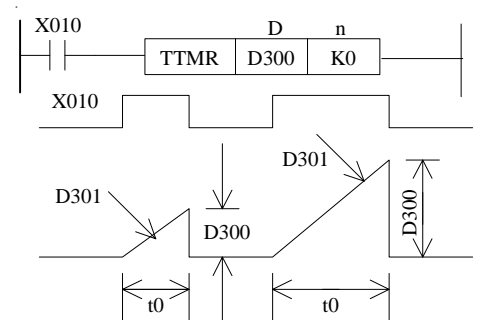
Tabela de dados		Valor do Contador S2+1
Elemento de dados	Preset de contagem para o contador S2	
D300	20	0
D301	30	1
D302	10	2
D303	40	3

3.7.5 TTMR (FNC 64)

Mnemônico	Função	Operandos		Passos do programa
		D	n	
TTMR FNC 64 (Monitoração de tempo)	Monitora a duração de um sinal e posiciona os dados temporizados num registro de dados	D Nota: 2 endereços de word são usados D e D+1	K, H ☆ Nota: n= 0: (D) = (D+1) X 1 n= 1: (D) = (D+1) X 10 n= 2: (D) = (D+1) X 100	TTMR: 5 passos

Operação:

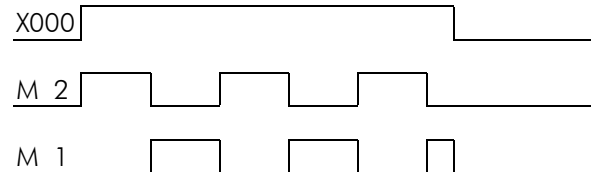
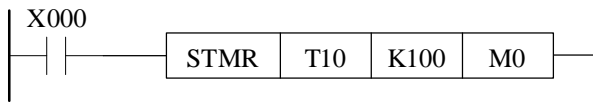
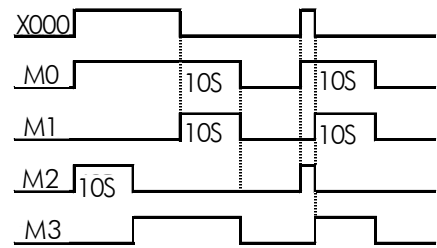
A duração de tempo que a instrução TTMR é habilitada é medida e armazenada no endereço D+1 (como uma contagem de períodos de 100ms).

O valor de dados de D+1 (em segs), multiplicado pelo fator selecionado pelo operando n, é movido para o registro D. O conteúdo de D poderia ser usado como os dados de origem para um habilitar um temporizador indireto ou até mesmo para manipulação de dados crus.

Quando a instrução TTMR é desabilitada, D+1 é automaticamente zerado (D não muda).

3.7.6 STMR (FNC 65)

Mnemônico	Função	Operandos			Passos do programa
		S	n	D	
STMR FNC 65 (Temporizador especial)	Fornece retardo na desenergização temporizadores de pulso	T Nota: Temporizadores 0 a 199 (dispositivos de 100mseg)	K, H ☆ Nota: n= 1 a 32.767	Y, M, S Nota: Usa 4 endereços consecutivos D+0 to D+3	STMR: 7 passos

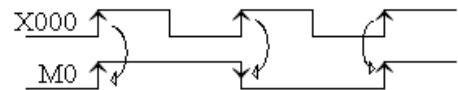
Operação:

Os temporizadores designados operarão por uma duração n com o efeito operacional sendo marcado pelos endereços D+0 a D+3. O endereço D+0 é um temporizador com retardo na desenergização, D+1 é um temporizador de pulso. Quando D+3 é usado na configuração abaixo, D+1 e D+2 agem numa seqüência de trem de pulsos alternada.

3.7.7 ALT (FNC 66)

Mnemônico	Função	Operandos	Passos do programa
		D	
ALT FNC 66 (inversão de Estado) ★	O estado do endereço designado é invertido a cada execução da instrução	Y, M, S	ALT, ALTP: 3 passos

Operação:

O status do endereço de destino (D) é alternado a cada operação da instrução ALT.

Isto significa que o status de cada endereço de bits irá alterar entre ON e OFF. Isto ocorrerá a cada varredura de programa a menos que uma habilitação por pulso ou um intertravamento seja usado.

A instrução ALT é ideal para trocar entre dois modos de operação, ex: iniciar e parar, ligar e desligar, etc.

3.7.8 RAMP (FNC 67)

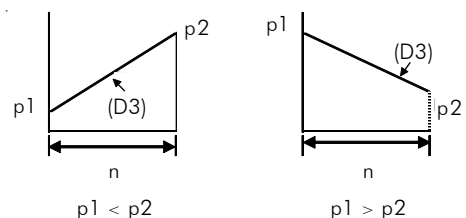
Mnemônico	Função	Operandos				Passos do programa
		S1	S2	D	n	
RAMP FNC 67 (Suaviza a variação do valor de um endereço)	Altera o valor de um endereço em rampa num número de passos definido	D Nota: O endereço D usa dois registros consecutivos identificados como D e D+1 estes são endereços de leitura somente.			K, H ✰ Nota: n= 1 a 32.767	RAMP: 9 passos

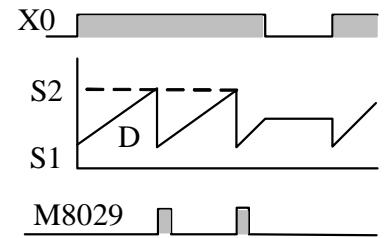
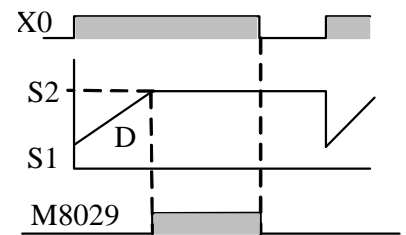
Operação:

A instrução RAMP varia um valor atual (D) entre os limites de dados programados pelo usuário (S1 e S2).

A 'jornada' entre estes limites extremos levam n varreduras de programa

. O número de varredura atual é armazenado no dispositivo D+1. Uma vez que o valor atual de D é igual ao valor programado do S2, o marcador M8029 é setado. A instrução RAMP pode variar, tanto aumentando quanto diminuindo diferenças entre S1 e S2.

p1 < p2 p1 > p2

Pontos a observar:

a) Usuários podem programar o modo de operação da instrução RAMP controlando o estado do marcador auxiliar M8026. Quando M8026 está desabilitado, a instrução RAMP estará no modo repetir, isto significa que o valor atual de D é igual a S2. A instrução RAMP irá zerar automaticamente e iniciar novamente, ou seja, o conteúdo de D será zerado para que o S1 e o endereço D+1 (o número atual de varreduras) sejam zerados. Isto é mostrado no diagrama ao lado.

Quando o M8026 é setado, a instrução RAMP operará no modo 'Hold'. Isto significa que uma vez que o valor atual de D é igual ao de S2, a instrução RAMP 'congelará' neste estado. Isto significa que M8029 estará setado enquanto a instrução permanecer habilitada e o valor de D não irá zerar até que a instrução seja reinicializada, ex: a instrução RAMP vai da posição OFF para ON novamente.

b) Se a instrução RAMP é interrompida antes de ser completada, então a posição atual da rampa é 'congelada' até que o sinal de habilitação seja restabelecido. Uma vez que a instrução RAMP é re-acionada, os registros D e D+1 zeram e o ciclo inicia do começo novamente.

c) Se a instrução RAMP é operada com um modo de varredura constante, ex: D8039 está escrito com o tempo de varredura desejado (um pouco mais longo do que o tempo de varredura atual) e M8039 está setado. Isto então permitiria um número n de varreduras (usados para criar a rampa entre S1e S2) ser associado a um tempo. Se uma varredura é igual ao conteúdo de D8039 então o tempo para completar a rampa é igual a n x D8039.

3.7.9 ROTC (FNC 68)

Mnemônico	Função	Operandos					Passos do programa
		S1	M1	M2	D	n	
ROTC FNC 68 (Monitor de rotação)	Monitora o sentido de giro e a posição de um encoder	D	K, H ✰ M1≥M2		D	K,H,D	ROTC: 9 passos

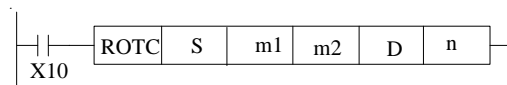
Operação:

A instrução ROTC é usada para auxiliar monitoração do sentido de giro e posição de um encoder.

Pontos a observar:

a) Esta instrução tem muitos dispositivos definidos automaticamente. Estes estão listados ao lado direito desta página.

b) A instrução ROTC só pode ser usada UMA vez.

c) A instrução ROTC usa um contador interno de 2 canais para detectar a direção do movimento e a distância percorrida.

Endereços associados à instrução

Endereços selecionados pelo usuário indireto:

D+0 Sinal do contador canal A - entrada

D+1 Sinal do contador canal B - entrada

D+2 Posição zero - entrada

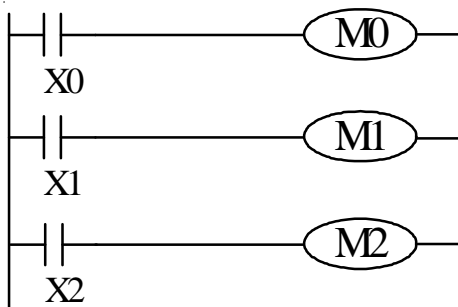
D+3 Alta velocidade para frente - saída

D+4 Baixa velocidade para frente - saída

D+5 Parar - saída

D+6 Baixa velocidade reverso - saída

D+7 Alta velocidade reverso - saída

Endereços D+0 e D+1 são usados para entrar os pulsos dos canais enquanto o endereço D+2 é usado para entrar a 'posição zero'. Estes endereços deveriam ser programados como mostrado no exemplo abaixo (onde a terminação física acontece nas entradas X associadas).

Constantes:

m1 Número de pulsos por volta do encoder

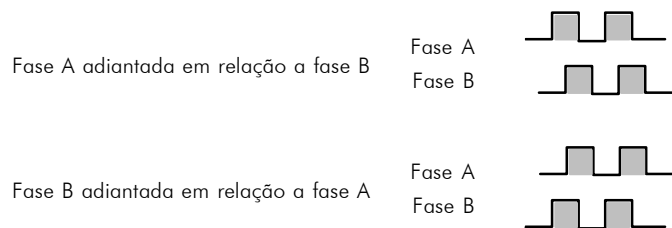
m2 Distância a ser percorrida em baixa velocidade (em pulsos do encoder)

Variáveis de operação:

S+0 Posição atual no 'ponto zero' SOMENTE LEITURA

S+1 Posição de destino (estação selecionada para onde deverá ser movida) relativo ao 'ponto zero' – Definido pelo usuário

S+2 Posição de início (estação selecionada para onde deverá ser movida) relativo ao 'ponto zero' – Definido pelo usuário

A direção do movimento é encontrada verificando a relação dos dois canais do contador de 2 canais, ex:

d) Quando a entrada 'ponto zero' (D+2) é recebida, o conteúdo do endereço S+0 é zerado. Antes de começar qualquer nova operação é recomendável garantir que o sistema está sempre parando no ponto zero. Sugere-se que seja feita uma marca na posição 'zero' a fim de verificar a repetibilidade. Isto poderia ser considerado como uma técnica de calibração. A recalibração deve ser executada periodicamente para garantir uma operação precisa.

e) Endereços D+3 à D+7 são automaticamente definidos pela instrução ROTC durante sua operação. Estes são usados como marcadores para indicar a operação que deveria ser executada em seguida.

f) Todas as posições são entradas na forma dos pulsos do encoder. Isto pode ser visto no exemplo a seguir:

Exemplo:

Uma mesa giratória tem um encoder que gera 400 (m1) pulsos por revolução. Há 8 posições (0 a 7) na mesa giratória, isto significa que, quando a mesa giratória se move de uma posição para a outra imediatamente a seguir, 50 pulsos de codificador são contados. Para mover o item localizado na posição 7 para a posição 3, os seguintes valores devem ser escritos na instrução ROTC:

S+1=3 x 50 = 150 (posição 3 em pulsos do encoder do ponto zero)

S+2=7 x 50 = 350 (posição 7 em pulsos do encoder do ponto zero)

m1 = 400 (número total de pulsos do encoder por volta)

A mesa giratória precisa se aproximar da posição de destino numa velocidade baixa, começando em 1,5 posições antes do destino. Portanto, m2 =1,5 x 50 = 75, distância de baixa velocidade nos dois lados da posição de destino (em pulsos de encoder)

3.7.10 SORT (FNC 69)

Mnemônico	Função	Operandos					Passos do programa
		S1	M1	M2	D	n	
SORT FNC 69 (Seleção de dados numa tabela)	Dados numa tabela definida podem ser selecionados em campos selecionados mantendo integridade do registro	D ✰	K, H,D ✰ Nota: m1= 1 a 32 m2= 1 a 6		D ✰	K, H D ✰Nota: n = 1 a m2	SORT: 11 passos

Operação:

Esta instrução constrói uma tabela de dados com m_1 linhas e m_2 colunas, tendo um início ou endereço inicial definido no parâmetro S. Quando a função é ativada, os dados da coluna selecionada em n são sorteados em ordem crescente, mantendo a integridade das linhas originais. A nova tabela de dados resultante dessa operação é armazenada no endereço de destino D.

Pontos a observar:

a) A organização dos dados é feita de forma crescente, a partir da coluna selecionada em n, porém sempre mantendo os demais dados da linha original;

b) As áreas de origem (S) e destino (D) podem ser as mesmas, MAS se as áreas são escolhidas para serem diferentes, não deveria haver nenhuma sobreposição entre as áreas ocupadas pelas tabelas.

c) Uma vez que a operação SORT foi completada o marcador M8029 é setado. Para uma seleção completa de uma tabela de dados, a instrução SORT será processada m1 vezes.

d) Durante uma operação SORT, os dados na tabela SORT não devem ser mudados. Se os dados são mudados, isto pode resultar numa tabela incorreta.

e) A instrução SORT só pode ser usada UMA vez no programa.

Exemplo:

Enquanto a entrada X21 estiver ativa, a instrução SORT será executada e organizará os dados conforme a coluna selecionada em n. Note que os endereços de fonte e destino são os mesmos.

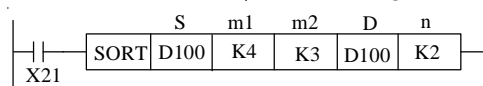
X21		S	m1	m2	D	n
⊣⊢	SORT	D100	K4	K3	D100	K2

Do exemplo de instrução e da 'tabela de dados' abaixo, a seguinte manipulação de dados ocorrerá quando programado no campo identificado.

Tabela de dados original:

	1	2	3
1	D100 32	D104 162	D108 4
2	D101 74	D105 6	D109 200
3	D102 100	D106 80	D110 62
4	D103 7	D107 34	D111 6

$m_1 = 4 \rightarrow$ 4 linhas

$m_2 = 3 \rightarrow$ 3 Colunas

a) n = 2 →Os dados da coluna 2 são organizados em ordem crescente:

	1	2	3
2	D100 74	D104 6	D108 200
4	D101 7	D105 34	D109 6
3	D102 100	D106 80	D110 62
1	D103 32	D107 162	D111 4

Ordem crescente

b) n = 1 → Os dados da coluna 1 são organizados em ordem crescente:

	1	2	3
4	D100 7	D104 34	D108 6
1	D101 32	D105 162	D109 4
2	D102 74	D106 6	D110 200
3	D103 100	D107 80	D111 62

Ordem crescente

3.8 DISPOSITIVOS EXTERNOS I/O - FUNÇÕES 70 À 79

Conteúdo:

TKY -	Conversão de dez endereços de bit em valor decimal	FNC 70
HKY -	Entrada Hexadecimal	FNC 71
DSW -	Multiplexação de entradas digitais (Chave "Thumbwheel")	FNC 72
SEGD -	Display de Sete Segmentos	FNC 73
SEGL -	Display de Sete Segmentos com memória	FNC 74
ARWS -	Setas de função para alteração de valores	FNC 75
ASC -	Conversão para código ASCII	FNC 76
PR-	Envio de dados para dispositivos de saída (impressão)	FNC 77

Lista de símbolos:

D – Endereço de destino.

S – Endereço de origem.

m, n - Número de endereços ativos, bits ou uma constante operacional.

Sufixos numéricos adicionais serão somados se houver mais de um operando com a mesma função e.g. D1, S3 ou para dispositivos listados/tabelados D_3+_0, $S+_9$ etc.

MSB – Bit mais significativo, às vezes usado para indicar o sinal matemático de um número, i.e. positivo = 0 e negativo = 1.

LSB – Bit menos significativo

Modificações das instruções:

□□□- Instrução de 16 bit, onde □□□ identifica o mnemônico da instrução.

□□□P - Instrução de 16 bits habilitada por um pulso (único).

D□□□ - Instrução de 32 bits.

D□□□P – Instrução de 32 bits habilitada por um pulso (único).

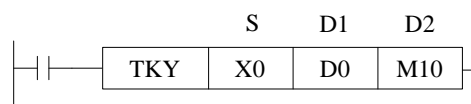
★ – Uma instrução repetitiva que irá mudar o valor de destino em cada varredura se não for habilitada por pulso.

☆– Um operando que não pode ser indexado ou a adição de V ou Z é inválida ou não terá efeito sobre o valor do operando.

3.8.1 TKY (FNC 70)

Mnemônico	Função	Operandos			Passos do programa
		S	D1	D2	
TKY FNC 70 (Leitura de dez bits consecutivos)	Lê 10 endereços com valores decimais associados e os armazena em um único número	X, Y, M, S Nota: utiliza 10 endereços consecutivos (identificados como S+0 a S+9)	KnY, KnM, KnS, T, C, D, V, Z Nota: utiliza 2 endereços consecutivos para operação de 32 bits	Y, M, S Nota: utiliza 11 endereços consecutivos (identificados como D2+0 a D2+10)	TKY: 7 passos DTKY: 13 passos

Operação:

Esta instrução pode ler de 10 endereços consecutivos (S+0 a S+9), converte o valor dos bits lidos em um valor numérico e armazenar este valor no endereço D1.

Pontos para observar:

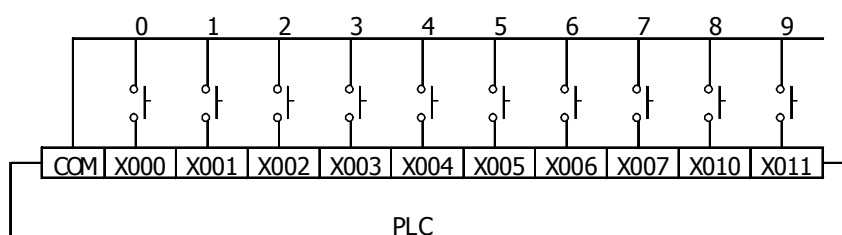
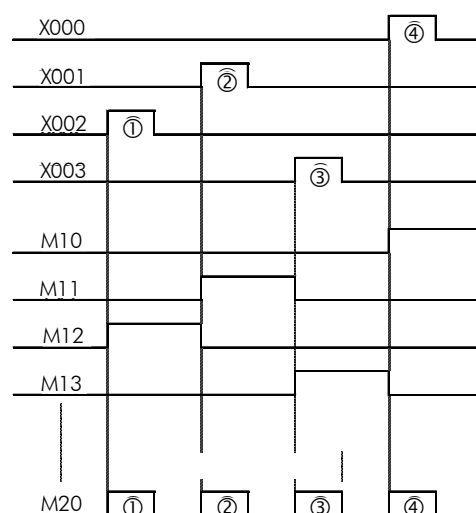
a) Quando um endereço de origem torna-se ativo, seu respectivo endereço de destino (bit) D2 também se torna ativo. Este endereço de destino permanecerá ativo até que um outro endereço de origem seja operado. Cada endereço de origem é associado diretamente para seu respectivo endereço D2, por exemplo, S+0 é endereçado para D2+0, S+7 endereçado para D2+7 etc. Estes, por sua vez, são convertidos automaticamente para um valor decimal que é então armazenado no endereço de destino especificado por D1.

b) Um endereço de origem poderá estar ativo a qualquer momento. O endereço de destino D2+10 é utilizado para significar que um bit (um dos 10 endereços de origem) foi alterado. D2+10 irá permanecer ativo pelo tempo que o bit ficar em nível alto (ON). Quando a instrução TKY estiver ativa, todo bit setado adiciona aquele dígito ao número armazenado em D1. Quando TKY estiver desabilitado, todos os endereços D2 são zerados, mas o valor dos dados em D1 mantém-se intactos.

c) Quando a instrução TKY é utilizada com operação de 16 bits, D1 pode armazenar números de 0 a 9.999, ou seja, no máximo 4 dígitos. Quando a instrução DTKY é utilizada (operação de 32 bits) valores de 0 a 9.999.999 (máx. 8 dígitos) podem ser acomodados em dois endereços consecutivos D1 e D1+1.

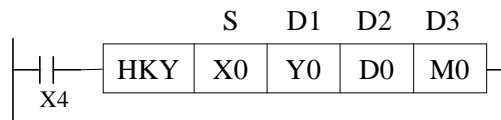
Em ambos os casos, se o número a ser armazenado exceder as faixas permissíveis, os dígitos mais altos irão ser desprezados até que um número permissível seja alcançado. Os dígitos que excederem os limites são perdidos e não poderão mais ser acessados pelo usuário. Não serão acrescidos zeros nos números convertidos, ou seja, 0127 será armazenado somente como 127.

d) A instrução TKY somente poderá ser utilizada UMA VEZ.

e) Usando a instrução acima como um breve exemplo: Se as 'entradas' identificadas (0) a (3) forem pressionadas nesta ordem, o número 2.130 será armazenado em D1. Se a tecla identificada como (5) for pressionada depois, o valor em D1 torna-se 1.309. A inicial '2' foi perdida.

3.8.2 HKY (FNC 71)

Mnemônico	Função	Operandos				Passos do programa
		S	D1	D2	D3	
HKY FNC 71 (Entrada da tecla Hexadecimal)	Multiplexa entradas e saídas para criar um teclado numérico com 6 teclas de função	X, Nota: utiliza 4 endereços consecutivos	Y, Nota : utiliza 4 endereços consecutivos	T, C, D, V, Z	Y, M, S	HKY: 9 passos DHKY: 17 passos

Operação 1 – Padrão:

Esta instrução multiplexa 4 saídas (D_1) e 4 entradas (S) para ler em 16 endereços diferentes. Valores decimais de 0 a 9 podem ser armazenados, enquanto 6 outros marcadores de função podem ser ajustados

Pontos para observar:

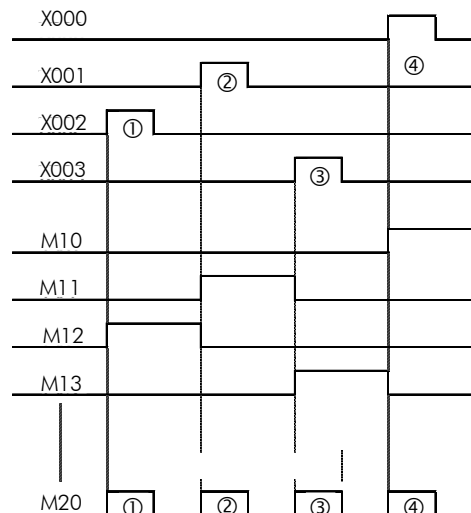
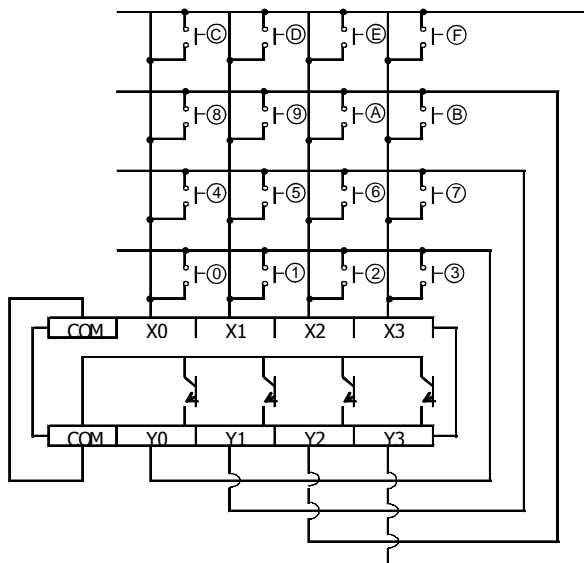
a) Cada um dos 10 primeiros endereços de origem multiplexados (identificados como 0 à 9) são endereçados diretamente para valores decimais 0 a 9. Quando um valor de entrada, por exemplo, um endereço de origem é ativado, então seu valor decimal associado é adicionado ao valor atualmente armazenado em D2. A ativação de qualquer bit setará o bit D_{3+7} enquanto aquele bit estiver setado.

b) Os últimos endereços de origem multiplexados (identificados como teclas de função A a F) são utilizados para ajustar endereços de bit D_{3+0} a D_{3+5} respectivamente. Estes marcadores de bit, uma vez setados, permanecem ON até que o próximo bit da função seja ativado. A ativação de qualquer destes bits setará o endereço de bit D_{3+6} a entrar na posição ON enquanto aquele bit for setado.

c) Em todos os casos de entrada de bits, quando dois ou mais bits forem setados, somente o primeiro ativado é que será considerado. Quando o pressionar de uma tecla é sentido, o marcador M8029 é setado. Quando a instrução HKY estiver desabilitada, todos os endereços D_3 são zerados, mas o valor de dado D_2 permanece em tacto.

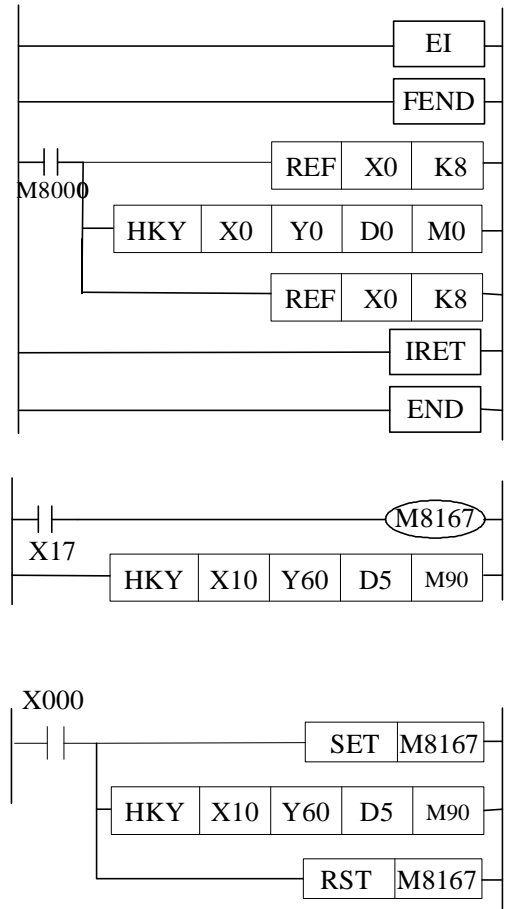
d) Quando a instrução HKY é utilizada com operação de 16 bits, D1 pode armazenar números de 0 a 9.999, ou seja, no máximo 4 dígitos. Quando a instrução DTKY é utilizada (operação de 32 bits) valores de 0 a 9.999.999 (máx. 8 dígitos) podem ser acomodados em dois endereços consecutivos D_1 e D_{1+1}.

Em ambos os casos, se o número a ser armazenado exceder as faixas permissíveis, os dígitos mais altos irão ser desprezados até que um número permissível seja alcançado. Os dígitos que excederem os limites são perdidos e não poderão mais ser acessados pelo usuário. Não serão acrescidos zeros nos números convertidos, ou seja, 0127 será armazenado somente como 127. Esta operação é similar ao da instrução TKY.

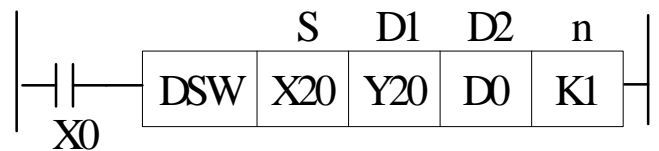
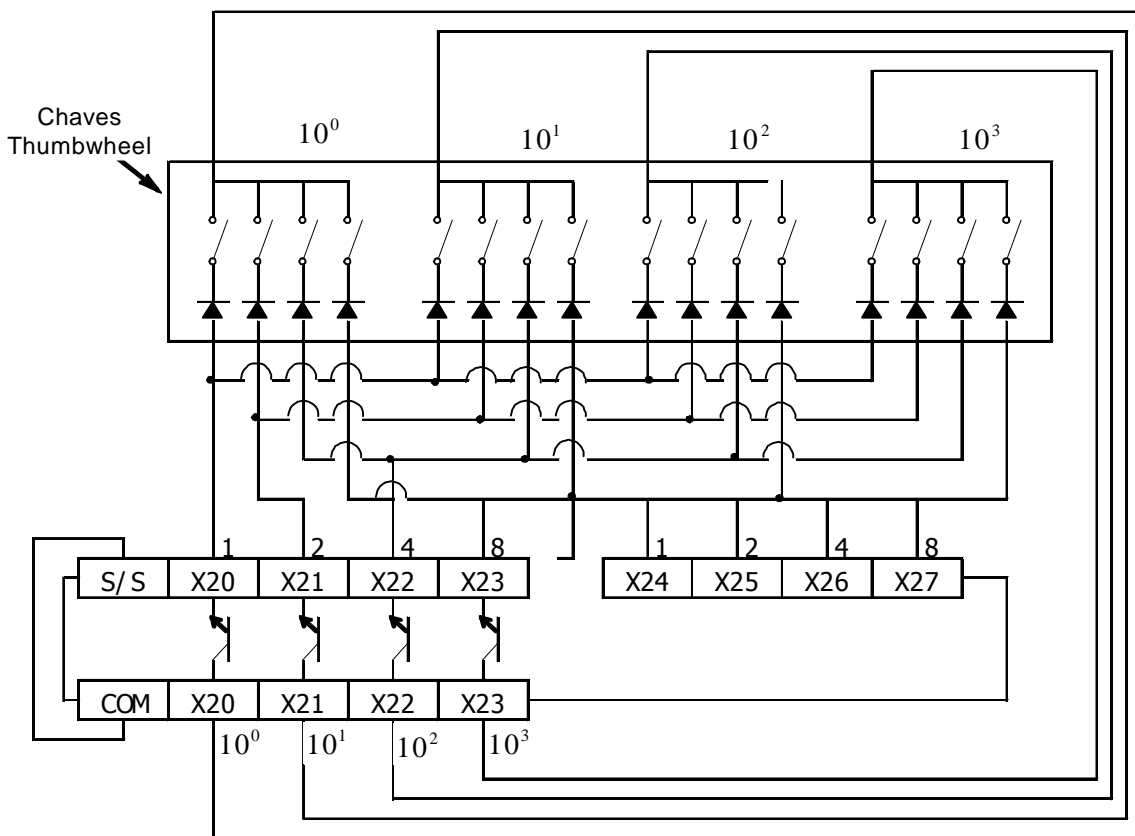
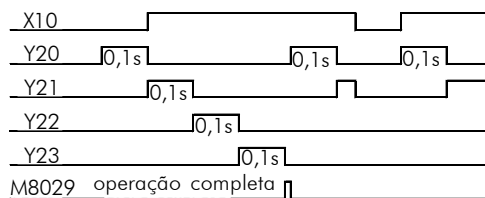
e) A instrução HKY somente poderá ser usada UMA VEZ.

f) Esta operação normalmente requer 8 varreduras para ler as entradas de teclas. Para alcançar um desempenho estável e que possa ser repetido, o modo de varredura constante deve ser usado, ou seja, o marcador M8039 deveria estar setado e um tempo de varredura definido pelo usuário estar escrito no registrador D8039. Porém, para uma resposta mais rápida, a instrução HKY deve ser programada em uma rotina de interrupção do temporizador como visto no exemplo oposto.

Operação 2 – Usando a instrução HKY com M8167:
Quando a instrução HKY é utilizada com o marcador M8167 LIGADO (visto à direita), a operação das teclas A à F permitem entrada real dos valores Hexadecimais de A a F respectivamente no endereço de dados D2. Isto é um acréscimo as teclas padrão de 0 a 9. Qualquer outra operação é igual o especificado em 'Operação 1 – Padrão'.

Os valores máximos de armazenagem para esta operação tornam-se FFFF em modo de 16 bits e FFFFFFFF em modo de 32 bits (double word).

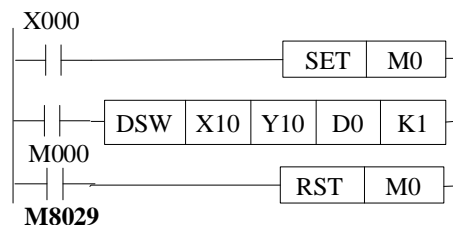
Estes dois exemplos de programa executam a mesma tarefa

3.8.3 DSW (FNC 72)

Mnemônico	Função	Operandos				Passos do programa
		S	D1	D2	n	
DSW FNC 72 (Multiplexação de entradas digitais)	Multiplexa a leitura de n conjuntos de entradas digitais (BCD) de uma chave "thumbwheel"	X Nota: Se n=2 então 8 endereços serão usados ao invés de 4.	Y Nota: utiliza 4 endereços consecutivos	T, C, D, V, Z Nota: Se n=2 então 2 endereços serão usados ao invés de 1	K, H ☆ Nota: n= 1 ou 2	DSW: 9 passos

Operação:
Esta instrução multiplexa 4 saídas (D1) através de 1 ou 2(n) conjuntos de chaves. Cada conjunto de 4 chaves consiste em uma "thumbwheel" providenciando uma entrada digital única.

Pontos para observar:
a) Quando n=1, somente 1 conjunto de switches é lido. A multiplexação é feita ligando as entradas da chave "thumbwheel" em paralelo de volta para 4 entradas consecutivas a partir de endereço de origem especificado no operando S. A leitura de dados (4 dígitos) é armazenada no endereço de memória D_2.

b) Quando n=2, dois conjuntos de chaves são lidas. Esta configuração requer 8 entradas consecutivas cujo primeiro endereço é definido no operando S. Os dados do primeiro conjunto de chaves, por exemplo, aqueles usando as primeiras 4 entradas, são lidos para o endereço de dados D_2. Os dados do segundo conjunto de chaves (novamente 4 dígitos) são lidos pelo endereço D_{2+1}.

c) As saídas utilizadas para multiplexação (D_1) serão acionadas ciclicamente enquanto a instrução DSW estiver habilitada. Após o término de uma leitura, o marcador M8029 é setado. O número de saídas **não** depende do número de chaves n.

d) Se a instrução DSW for suspensa no meio da operação, quando for reiniciada irá partir do começo de seu ciclo e não do último status alcançado.

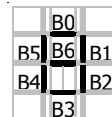
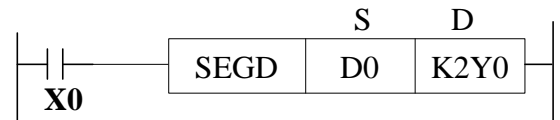
X000
SET M0
DSW X10 Y10 D0 K1
M000
RST M0
M8029

e) É recomendado que unidades de saída de transistores sejam utilizadas com esta instrução. Porém, se a técnica de programação ao lado for utilizada, o módulo de saídas à relé pode ser usado com sucesso, pois as saídas não estarão continuamente ativas.

3.8.4 SEGD (FNC 73)

Mnemônico	Função	Operandos		Passos do programa
		S	D	
SEGD FNC 73 (Display de sete segmentos)	Dados Hex são decodificados para um formato utilizado nos displays de sete segmentos	K, H KnX, KnY, KnM, KnS, T, C, D, V, Z Nota: Utiliza somente os 4 bits mais baixos	KnY, KnM, KnS, T, C, D, V, Z Nota: Os 8 bits mais altos permanecem inalterados	SEGD, SEGDP: 5 passos

Operação:

Um dígito hexadecimal (0 a 9, A a F) ocupando os 4 bits mais baixos do endereço de origem S é decodificado para um formato de dados utilizado num display de sete segmentos. Uma representação do dígito hex é então apresentada. Os dados do decodificador são armazenados nos 8 bits mais baixos do endereço de destino D. Os 8 bits mais altos do mesmo endereço são desprezados. O desenho ao lado mostra o controle de bit do display de sete segmentos. Os LEDs ativos correspondem aos bits setados dos 8 bits mais baixos do endereço de destino D.

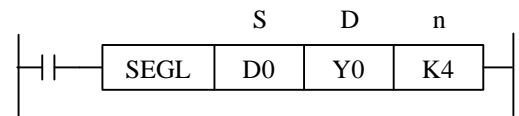
B0
B5 B6 B1
B4 B2
B3

Pode ser observado que B7 NÃO é utilizado. Então B7 do endereço de destino D sempre estará na desenergizado.

3.8.5 SEGL (FNC 74)

Mnemônico	Função	Operandos			Passos do programa
		S	D	n	
SEGL FNC 74 (Display de Sete segmentos com memória)	Escreve dados para um display de um conjunto de endereços – 4 dígitos por conjunto, máx. 2 conjuntos	K, H KnX, KnY, KnM, KnS T, C, D, V, Z	Y Nota: n = 0 à 3, 8 saídas são utilizadas n = 4 à 7, 12 saídas são utilizadas	K, H,☆ Nota: n= 0 à 3, 1 conjunto de 7 Seg ativo= 4 a 7, 2 conjuntos de 7 Seg ativo	SEGL: 7 passos

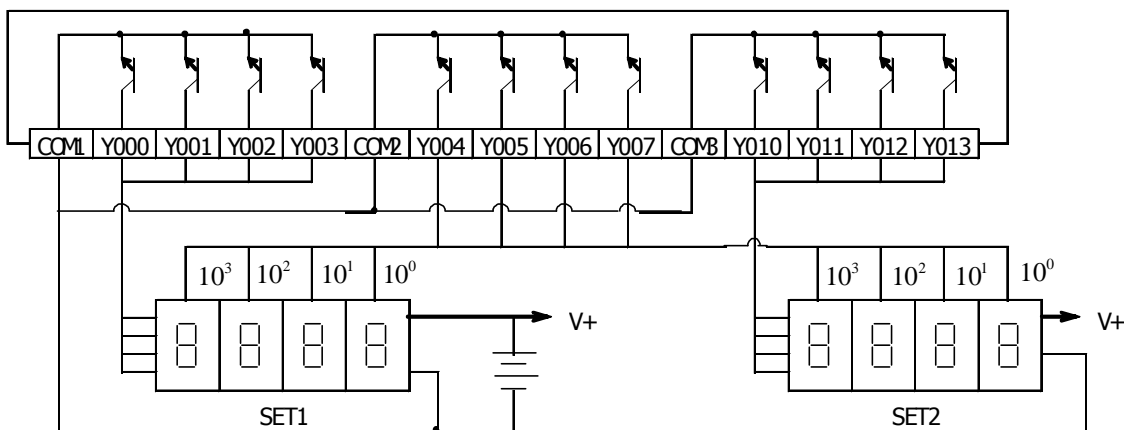
Operação:

Esta instrução pega um valor decimal de origem (S) e o escreve para um conjunto de 4 saídas (D) multiplexadas.

Devida a variação entre fabricantes da lógica utilizada com displays de sete segmentos com memória, esta instrução pode ser modificada para se adequar à maioria dos requisitos de lógica. Configurações são selecionadas dependendo do valor de n, ver abaixo.

Pontos para observar:

a) Dados são escritos para um conjunto de saídas multiplexadas (D+0 a D+7, 8 saídas) e, portanto, num display de sete segmentos. Um conjunto de displays consiste em 4 unidades de sete segmentos que formam 1 dígito cada. No máximo dois conjuntos de displays podem ser habilitados com esta instrução. Quando dois conjuntos são utilizados, os displays dividem as mesmas saídas de atualização (D+4 a D+7 são as saídas de atualização). Um conjunto adicional de 4 endereços de saída é necessário para fornecer os novos dados para o segundo conjunto de displays (D+10 a D+13, esta é uma adição octal). As saídas de atualização fazem os dados escritos serem memorizados no display de sete segmentos.

b) Os dados de origem dentro da faixa de 0 a 9.999 (decimal) são escritos para as saídas mutiplexadas. Quando um conjunto de displays é utilizado, estes dados são tirados do endereço especificado como operando S. Quando dois conjuntos de displays estão ativos, o endereço de origem S+1 fornece os dados para o segundo conjunto de displays. Estes dados devem novamente estar entre a faixa de 0 a 9.999. Quando utilizando dois conjuntos de displays, os dados são tratados como **dois** números separados e não são combinados para fornecer uma saída única de 0 a 99.999.999.

c) A instrução SEGL leva 12 varreduras de programa para completar um ciclo de saída, independente do número de conjuntos de display utilizados.

d) Se a instrução SEGL for desabilitada no meio da operação, quando for reiniciada irá partir do começo de seu ciclo e não do último status alcançado.

e) O CLP pode executar no máximo DUAS instruções SEGL.

Selecionando o valor correto para o operando n

A seleção do parâmetro n depende de 4 fatores;

1) O tipo e lógica utilizados para a saída do CLP
2) O tipo e lógica utilizados para as linhas de dados de sete segmentos
3) O tipo e lógica utilizados para o sinal de atualização dos valores no display
4) Quantos conjuntos de displays serão utilizados

Dispositivo considerado		Lógica positiva	Lógica negativa
Lógica do CLP		CLP, ALTA, V+, Pull-up resistor, 0V	CLP, BAIXA, V+, Pull-up resistor, 0V
		Com uma saída de origem, quando a saída for ALTA a lógica interna é '1'	Com uma saída NPN, quando a saída for BAIXA a lógica interna é '1'
Lógica do display de sete segmentos	Lógica do sinal de atualização	Dados são lidos e armazenados quando este sinal é setado, ou sua lógica é '1'	Dados são lidos e armazenados quando este sinal for BAIXO, ou sua lógica é '1'
	Lógica do sinal dos dados	Linhas de dados ativos são armazenadas no display quando em nível ALTO, ou eles tem um valor de lógica de '1'	Linhas de dados ativos são armazenadas no display quando em nível BAIXO, ou eles tem um valor de lógica de '1'

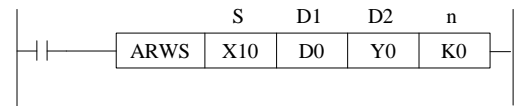
Há dois tipos de sistemas de lógica disponíveis, lógica positiva e negativa. Dependendo do tipo do sistema, o valor de n pode ser selecionado da tabela abaixo com a referência final para o número de conjuntos de displays de sete segmentos sendo utilizados:

Lógica do CLP	Lógica do display de sete segmentos		n	
	Lógica dos dados	Lógica de atualização	1 conjunto de displays	2 conjuntos de displays
Positivo (PNP)	Positivo (Alto)	Positivo (Alto)	0	4
Negativo (NPN)	Negativo (Baixo)	Negativo (Baixo)		
Positivo (PNP)	Positivo (Alto)	Negativo (Baixo)	1	5
Negativo (NPN)	Negativo (Baixo)	Positivo (Alto)		
Positivo (PNP)	Positivo (Alto)	Negativo (Baixo)	2	6
Negativo (NPN)	Negativo (Baixo)	Positivo (Alto)		
Positivo (PNP)	Positivo (Alto)	Positivo (Alto)	3	7
Negativo (NPN)	Negativo (Baixo)	Negativo (Baixo)		

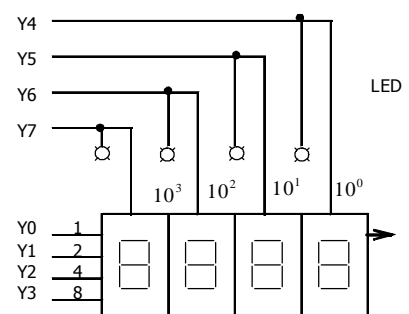
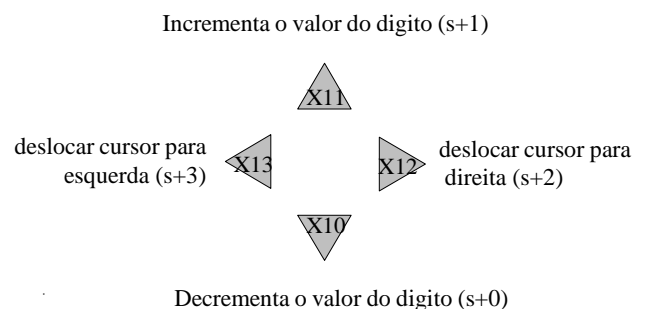
3.8.6 ARWS (FNC 75)

Mnemônico	Função	Operandos				Passos do programa
		S	D1	D2	n	
ARWS FNC 75 (Setas de função para alteração de valores)	Cria um painel de entrada de dados numéricos definidos por um usuário (4 teclas)	X, Y, M, S Nota: utiliza 4 endereços consecutivos	T, C, D, V, Z Nota: dados são armazenados em um formato decimal	Y Nota: utiliza 8 endereços consecutivos	K, H ☆ Nota: n= 0 a 3	ARWS: 9 passos

Operação:

Esta instrução apresenta o conteúdo de um único endereço de dados D1 em um conjunto de displays de sete segmentos com 4 dígitos. Os dados dentro de D1 estão na verdade em um formato de decimal padrão, mas são automaticamente convertidos para BCD para serem apresentados nos displays de sete segmentos. Cada dígito do número apresentado pode ser selecionado e editado. O procedimento de editoração muda diretamente o valor do endereço especificado como D1.

Pontos para observar:

a) Os dados armazenados no endereço de destino D1 podem ter um valor na faixa 0 a 9.999 (decimal), dados de 4 dígitos. Cada dígito pode ser incrementado (S+1) ou decrementado (S+0) pressionando as teclas de controle associadas. Os números editados automaticamente alteraram de 9-0-1 e 1-0-9 quando as teclas forem pressionadas. O dado de dígito é apresentado pelos 4 endereços mais baixos de D2, D2+0 a D2+3.

b) Na ativação inicial da instrução ARWS, o dígito na posição numérica 10^3 é atualmente selecionado. É possível navegar nas posições dos dígitos seqüencialmente indo para a esquerda (S+2) ou para a direita (S+3). Quando o último dígito for alcançado, a instrução ARWS automaticamente retorna a posição inicial, ou seja, após a posição 10^3, a posição 10^o é selecionada e vice-versa. Cada dígito é fisicamente selecionado por uma saída de atualização diferente.

c) Para ajudar o usuário de painel de controle com instrução ARWS, lâmpadas adicionais podem ser ligadas em paralelo com as saídas de atualização de cada dígito. Isto indicaria qual dígito foi selecionado para editoração.

d) O parâmetro n tem a mesma função que o parâmetro n da instrução SEGL – ver item 3.8.5, 'Selecionando o valor correto para o operando n'. Nota: como a instrução ARWS só controla um conjunto de displays, somente valores de 0 a 3 são válidos para n.

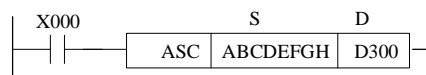
e) A instrução ARWS pode ser utilizada UMA VEZ. Esta instrução somente deveria ser utilizada em CLPs de saída de transistor.

3.8.7 ASC (FNC 76)

Mnemônico	Função	Operandos		Passos do programa
		S	D	
ASC FNC 76 (Conversão de código ASCII	Uma string com dados alfanuméricos pode ser convertida para seu código ASCII	Dados alfanuméricos Ex. 0-9, A - Z e a - z etc. Nota: Só uma string de 8 caracteres pode ser editada por vez.	T, C, D Nota: utiliza 4 endereços consecutivos	ASC: 7 passos

Operação:

A string de dados de origem S consiste em 8 caracteres tirados do conjunto de caracteres (Char) imprimíveis ASCII. Se menos que 8 caracteres forem usados, os espaços vazios serão preenchidos com zeros (ASCII 00).

X000		S	D
─┤├─	ASC	ABCDEFGH	D300

Os dados de origem são convertidos para seus códigos ASCII associados. Os códigos são então armazenados no endereço de origem D, veja exemplo abaixo:

D	Byte	
	Alto	Baixo
D300	42 (B)	41 (A)
D301	44 (D)	43 (C)
D302	46(F)	45 (E)
D303	48 (H)	47 (G)

Nota:

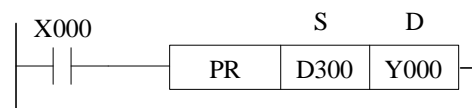
☑ Caracteres ASCII **não podem** ser editados com um programador portátil.

☑ Quando = ON, somente os 8 bytes mais baixos do endereço D podem estar disponíveis para armazenar dados. Os 8 bytes mais altos serão escritos com 0.

3.8.8 PR (FNC 77)

Mnemônico	Função	Operandos		Passos do programa
		S	D	
PR FNC 77 (Imprimir)	Envia para dispositivos de saída como displays ou impressoras dados ASCII	T, C, D Nota: modo de 8 bytes (M8027= na posição OFF) utiliza 4 endereços consecutivos e modo de 16 bytes (M8027= LIGADO) utiliza 8 endereços consecutivos	Y Nota: utiliza 10 endereços consecutivos	PR: 5 passos

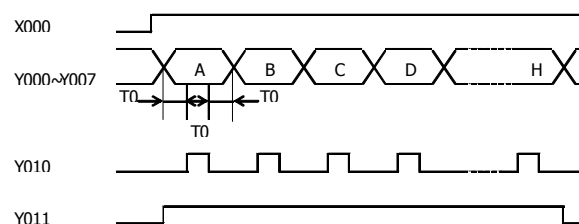
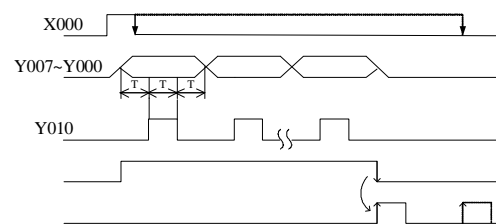
Operação:

Dados de origem (armazenados como valores ASCII) são lidos byte a byte pelos endereços de dados de origem. Cada byte é endereçado diretamente para os primeiros 8 endereços de destino consecutivos (D+0 to D+7). Os últimos dois bits de destino fornecem um sinal de atualização (D+10, numerado em octal) e um marcador de "execution/busy" (D+11, em octal).

Pontos para observar:

a) Os dados do byte de origem endereçam o bit mais baixo para o primeiro endereço de destino D+0. Conseqüentemente o bit mais alto do byte é mandado para o endereço de destino D+7.

b) A instrução PR pode ser utilizada UMA VEZ.

c) A operação da instrução PR dependente da varredura do programa. Sob circunstâncias padrão leva 3 varreduras de programa para enviar 1 byte. Porém, para uma operação mais rápida a instrução PR poderia ser escrita numa rotina de interrupção habilitada por tempo similar àquela demonstrado para HKY no item 3.8.2.

d) Operações de 8 bytes têm o seguinte diagrama de tempos. Deve ser observado que quando a entrada do inversor (no exemplo X0) estiver desabilitada, a instrução PR irá cessar sua operação. Quando for re-iniciada, a instrução PR irá iniciar do início da mensagem. Uma vez que todos os 8 bytes forem enviados, o marcador de "execution/busy" é desabilitado e a instrução PR suspende sua operação.

e) Operação de 16 bytes requer o marcador auxiliar especial M8027 para habilitação (é recomendado que M8000 seja utilizado como uma entrada do inversor). Neste modo de operação a entrada do inversor (no exemplo X0) não precisa estar ativa todo o tempo. Uma vez que a instrução PR for ativada, ela irá operar continuamente até que todos os 16 bytes de dados tenham sido enviados ou o valor 00H (nulo) tenha sido enviado. Uma vez completa a operação, o marcador "execution/busy" (D+11, octal) é desenergizado e M8029, o marcador de operação finalizada é setado.

3.9 DISPOSITIVOS EXTERNOS - FUNÇÕES 80 À 88

Conteúdo:

RS -	Comunicação Serial	FNC 80
PRUN -	Transferência Octal	FNC 81
ASCI -	Conversão Hexadecimal para ASCII	FNC 82
HEX -	Conversão ASCII para Hexadecimal	FNC 83
CCD -	Verificação da paridade de uma pilha	FNC 84
VRRD	Leitura de Volume	FNC 85
VRSC	Escala de Volume	FNC 86
MBUS -	Transmissão de Dados Seriais MODBUS	FNC 87
PID -	Controle PID	FNC 88

Lista de símbolos:

D – Endereço de destino.

S – Endereço de origem.

m, n - Número de endereços ativos, bits ou uma constante operacional.

Sufixos numéricos adicionais serão somados se houver mais de um operando com a mesma função e.g. D1, S3 ou para dispositivos listados/tabelados D3+0, S+9 etc.

MSB – Bit mais significativo, às vezes usado para indicar o sinal matemático de um número, ex: positivo = 0 e negativo = 1.

LSB – Bit menos significativo

Modificações das instruções:

☐☐☐ - Instrução de 16 bit, onde ☐☐☐ identifica o mnemônico da instrução.

☐☐☐P - Instrução de 16 bits habilitada por um pulso (único).

D☐☐☐ - Instrução de 32 bits.

D☐☐☐P - Instrução de 32 bits habilitada por um pulso (único).

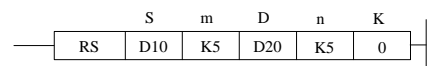
★ - Uma instrução repetitiva que irá mudar o valor de destino em cada varredura se não for habilitada por pulso.

☆ Um operando que não pode ser indexado ou a adição de V ou Z é inválida ou não terá efeito sobre o valor do operando.

3.9.1 RS (FNC 80)

Mnemônico	Função	Operandos					Passos do programa
		S	m	D	n	K	
RS FNC 80 (Instrução de Comunicação Serial)	Usado para controlar comunicações seriais de/para o controlador programável	D (incluindo registro de arquivos)	K, H, D ☆ m = 0 a 255	D	K, H, D ☆ m = 0 a 255	0,1	RS: 11 passos

Operação:

Tal instrução é utilizada para enviar ou receber dados em série sem protocolo, junto com os cartões de expansão opcionais RS-232, RS-485 ou pela porta embutida RS485.

Pontos para observar:

a) Esta instrução tem muitos endereços definidos automaticamente. Estes estão listados em 'Dispositivos de dados'.

b) A instrução RS tem duas partes, enviar (ou transmissor) e receber dados. Os primeiros elementos da instrução RS especificam o buffer de dados (S) da transmissão como um endereço principal, que contém m número de elementos em uma pilha seqüencial.

A especificação da área para armazenamento dos dados recebidos é definida nos últimos dois parâmetros da instrução RS. O destino (D) para mensagens recebidas tem um comprimento de buffer ou pilha de n elementos de dados. O tamanho dos buffers de envio e recebimento dita o tamanho de uma única mensagem. Tamanhos de buffers podem ser atualizados a cada transmissão:

1) Buffer de transmissão – antes que a transmissão ocorra, antes que M8122 seja setado;

2) Buffer de recepção – depois que uma mensagem foi recebida e antes que M8123 seja resetado.

c) Dados não podem ser enviados enquanto uma mensagem está sendo recebida, a transmissão será adiada – ver M8121.

d) Mais de uma instrução RS pode ser programada, mas somente uma poderá estar ativa, duas instruções não podem ser habilitadas ao mesmo tempo.

Dispositivos de dados:

a) Para porta RS485

1) Pronto para enviar (M8121): o marcador será setado quando a solicitação para envio dos dados é recebida. O marcador resetará automaticamente quando os dados forem enviados.

2) Solicitação de envio (M8122): Quando M8122 for setado pelo pulso pronto para enviar ou transmissão finalizada, a string de dados, que é (S), cujo comprimento é m, será enviada. M8122 será automaticamente resetado quando a transmissão for finalizada.

3) Final de envio: (M8123): M8123 será setado quando o envio estiver acabado. Favor zerar M8123 somente depois que os dados recebidos estiverem salvos em certos registros.

4) Falha na transmissão (M8129): Se nova tentativa de recebimento de dados não iniciar dentro do tempo especificado, o marcador de falha na transmissão será setado. Quando o envio terminar, M8123 será zerado e M8129 irá automaticamente resetado.

5) Frame de comunicação (D8120): refere-se ao frame da instrução MBUS.

6) Numero de dados restantes a serem enviados (D8122)

7) Numero de dados recebidos (D8123)

8) Tempo de watchdog para informar falha na transmissão (D8129): tempo de watchdog para falha de comunicação (5～255) x10ms.

b) Para placa de expansão RS485/ RS232

1) Pronto para enviar (M8321)

2) Solicitação de envio (M8322)

3) Final de envio (M8323)

4) Marcador de erro (M8124)

5) Falha na transmissão (M8329)

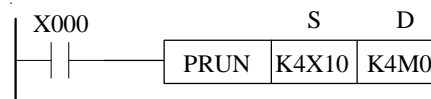
6) Frame de comunicação (D8320)

7) Numero de dados restantes a serem enviados (D8322)

3.9.2 PRUN (FNC 81)

Mnemônico	Função	Operandos		Passos do programa
		S	D	
PRUN FNC 81 (Transferência Octal)	Transmissão octal de bits	KnX, KnM	KnY, KnM	PRUN, PRUNP: 5 passos DPRUN, DPRUNP: 9 passos
		Nota: n = 1 a 8 Para facilidade e conveniência, o Bit do endereço principal deve ser um múltiplo de '10', por exemplo, X10, M1000, Y30 etc.		

Operação:

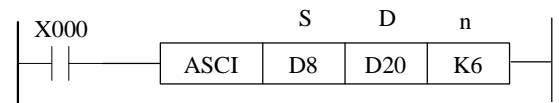
Esta instrução permite mover dados de origem para a área de transmissão de bits.

c) A instrução PRUN permite que dados sejam movidos para a área de transmissão de bits ou para fora da área de dados recebidos de bit. A instrução PRUN difere das instrução de movimentação de dados entre endereços de memória porque opera em octal. Isto significa que se K4X20 foi movido utilizando a instrução PRUN para K4M920, dados não seriam escritos para M928 e M929, sendo que estes endereços não são considerados na contagem octal. Isto pode ser visto no diagrama abaixo.

3.9.3 ASCI (FNC 82)

Mnemônico	Função	Operandos			Passos do programa
		S	D	n	
ASCI FNC 82 (Converte HEX para ASCII)	Converte um dados hexadecimal para ASCII	K, H, KnX, KnY, KnM, KnS T, C, D, V, Z	KnY, KnM, KnS, T, C, D	K, H Nota: n = 1 a 256 ?	ASCI, ASCIP: 7 passos

Operação:

Esta instrução lê n caracteres de dados hexadecimais de um endereço de origem (S) e os converte no código ASCII equivalente.

Isto é então armazenado no destino (D) para n numero de bytes.

Pontos para observar:

Favor observe que os dados são convertidos 'como lidos', ou seja, utilizando o exemplo acima com os seguintes dados em (D9, D8) ABCDH,EF26H. Pegando os primeiros caracteres (dígitos) hexadecimais n da direita (neste caso n=6) e convertendo eles para ASCI irá armazenar valores em 6 bytes consecutivos de D20, onde D20 = (67, 68), D21 = (69, 70) e D22 = (50, 54) respectivamente. Se estes símbolos fossem tratados como caracteres verdadeiros isto leria CDEF26.

Isto pode ser mostrado graficamente como na tabela abaixo. Favor observe que os dados de origem (S)são lidos do endereço mais significativo para o menos significativo, enquanto os dados de destino (D) são lidos na direção oposta.

A instrução ASCI pode ser utilizada com o M8161, marcador que define 8 bits/16 bits. O efeito deste marcador é exatamente igual. O exemplo mostra o efeito quando M8161 resetado. Se M8161 estivesse setado, somente o byte de destino mais baixo (b0-7) seria utilizado para armazenar os dados e então 6 registros de dados seriam necessários (D20 à D25).

Origem (S)		Dados
D9	b12-15	A
	b8-11	B
	b4-7	C
	b0-3	D
D8	b12-15	E
	b8-11	F
	b4-7	2
	b0-3	6

Destino (D)		Código ASCII		Símbolo
		HEX	DEC	
D20	b8-15	43	67	'C'
	b0-7	44	68	'D'
D21	b8-15	45	69	'E'
	b0-7	46	70	'F'
D22	b8-15	32	50	'2'
	b0-7	36	54	'6'

Códigos de Caracteres ASCII

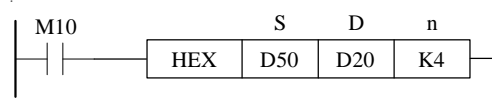
A tabela abaixo identifica os dígitos hexadecimais utilizáveis e seus respectivos códigos ASCII

Caractere HEX		0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
Código ASCII	HEX	30	31	32	33	34	35	36	37	38	39	41	42	43	44	45	46
	DEC	48	49	50	51	52	53	54	55	56	57	65	66	67	68	69	70
Símbolo do Caractere		'0'	'1'	'2'	'3'	'4'	'5'	'6'	'7'	'8'	'9'	'A'	'B'	'C'	'D'	'E'	'F'

3.9.4 HEX (FNC 83)

Mnemônico	Função	Operandos			Passos do programa
		S	D	n	
HEX FNC 83 (Converte ASCII em HEX)	Converte um dado ASCII para seu equivalente hexadecimal	K, H, KnX, KnY, KnM, KnS, T, C, D	KnY, KnM, KnS T, C, D, V, Z	K, H Nota: n = 1 a 256 ☆	HEX, HEXP: 7 passos

Operação:

Esta função lê n bytes de dados ASCII do endereço de origem (S) e os converte em caractere Hexadecimais equivalentes. Isto é armazenado no destino (D) para n número de bytes.

Pontos para observar:

Favor observar que esta instrução 'funciona em reverso' à instrução ASCI, os dados ASCII armazenados em bytes são convertidos em valores hexadecimais associados. A instrução HEX pode ser utilizada com o marcador de seleção de modo 8 bits/16bit, M8161. Neste caso os dados de origem (S) são lidos ou do byte mais baixo (8 bits) quando M8161 está setado, ou uma word inteira quando M8161 está resetado. Usando o exemplo acima com os seguintes dados nos endereços D50 e D51 respectivamente (43H,41H) (42H,31H) e supondo que M8161 está setado.

Os dados ASCI são convertidos para seus equivalentes hexadecimais e armazenados seqüencialmente, dígito por dígito no endereço de destino.

Se M8161 estivesse resetado, o conteúdo de D20 leria CAB1H.

Origem (S)		Código ASCII		Símbolo
		HEX	DEC	
D51	b8-15	43	67	'C'
	b0-7	41	65	'A'
D50	b8-15	42	66	'B'
	b0-7	31	49	'1'

Destino (D)		Dados
D20	b12-15	-
	b8-11	-
	b4-7	A
	b0-3	1

Para mais detalhes em relação ao uso da instrução HEX e sobre as faixas de dados ASCII disponíveis, favor veja o ponto de informações 'Códigos de Caracteres ASCII' na instrução ASCI da página anterior.

Importante:

Se tentarem acessar um código ASCII (HEX ou Decimal) que cai fora das faixas especificadas na tabela da página anterior, a instrução não será executada. O Erro 8067 é marcado no registro de dados D8004 e o erro 6706 é identificado em D8067. Cuidado deve ser tomado quando utilizando o marcador M8161, e mais ainda na especificação do número do elemento 'n' que será processado porque estes são os lugares mais prováveis que este erro será causado.

3.9.5 CCD (FNC 84)

Mnemônico	Função	Operandos S	Operandos D	Operandos n	Passos do programa
CCD FNC 84 (Verifica Código)	Verifica a paridade 'vertical' da pilha de dados	KnX, KnY, KnM, KnS T, C, D	KnY, KnM, KnS T, C, D	K, H D Nota: n = 1 a 256 ✰	CCD, CCDP: 7 passos

Operação:

Esta instrução olha uma pilha de bytes (8 bits) cujo endereço inicial é defino por (S) para n bytes e verifica a paridade do padrão vertical dos bits e soma a pilha total de dados. Estes dois dados são então armazenados no destino (D).

Pontos para observar:

a) A soma da pilha de dados é armazenada no destino D enquanto a paridade da pilha de dados é armazenada em D+1.

b) Durante a verificação de paridade, um resultado par é indicado pelo uso de um 0 (zero) enquanto uma paridade impar é indicada por um 1 (um).

c) Esta instrução pode ser utilizada com o marcador M8161 para definir operações de 8 bits/16 bits. Os seguintes resultados irão acontecer sob estas circunstâncias.

M8161=OFF										
Origem (S)			Padrão do Bit							
D100	H	FF	1	1	1	1	1	1	1	1
	L	FF	1	1	1	1	1	1	1	1
D101	H	FF	1	1	1	1	1	1	1	1
	L	00	0	0	0	0	0	0	0	0
D102	H	F0	1	1	1	1	0	0	0	0
	L	0F	0	0	0	0	1	1	1	1
Paridade vertical D1			0	0	0	0	0	0	0	0
SUM D0		3FC								

M8161=ON										
Origem (S)			Padrão do Bit							
D100	L	FF	1	1	1	1	1	1	1	1
D101	L	00	0	0	0	0	0	0	0	0
D102	L	0F	0	0	0	0	1	1	1	1
D103	L	F0	1	1	1	1	0	0	0	0
D104	L	F0	1	1	1	1	0	0	0	0
D105	L	0F	0	0	0	0	1	1	1	1
Paridade vertical D1			1	1	1	1	1	1	1	1
SUM D0		2FD								

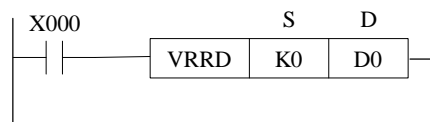
Deve ser notado que quando M8161 estiver setado, 'n' representa o número de bytes consecutivos verificados pela instrução CCD. Quando M8161 estiver na posição ON, somente os bytes mais baixos de 'n' palavras consecutivas serão utilizados.

A 'SUM' é simplesmente uma somatória da quantidade total de dados na pilha de dados. A paridade é verificada verticalmente pela pilha de dados como exibido pelas áreas sombreadas.

3.9.6 VRRD (FNC 85)

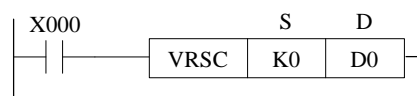
Mnemônico	Função	Operandos S	Operandos D	Passos do programa
VRRD FNC 85 (Ler Volume)	Lê o volume a partir de 2 potenciômetros, numerados No0, No1, e 6 VRs - potenciômetros na placa de expansão, numerada No2~No7.	K, H Nota: S= 0 a 7	KnY, KnM, KnS T, C, D, V, Z	VRRD, VRRDP: 5 passos

Operação:

O valor atual do potênciometro selecionado é lido, em formato de 10 bits, somente valores de 0 a 1023 são legíveis. Os dados lidos são armazenados no endereço de destino identificado sob o operando D.

3.9.7 VRSD (FNC 86)

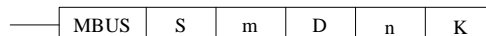
Mnemônico	Função	Operandos		Passos do programa
		S	D	
VRSC FNC 86 (Escala de Volume)	Lê escala (0~10) de 2 potenciômetros numerados No0, No1, e 6 potenciômetros na placa de expansão, numerada No2~No7.	K, H Nota: S= 0 a 7	KnY, KnM, KnS T, C, D, V, Z	VRSC, VRSCP: 5 passos

Operação:

O volume identificado (S) no CLP é lido como um potenciômetro com 11 posições ajustadas (0 a 10). Os dados de posição são armazenados no endereço D como número inteiro na faixa de 0 a 10.

3.9.8 MBUS (FNC 87)

Mnemônico	Função	Operandos					Passos do programa
		S	m	D	n	K	
MBUS FNC 87	Habilita comunicação MODBUS utilizando as portas da placa de comunicação RS485 / RS232	D	K,H,D m=0~255	D	K,H,D n=0~255	K,D 0,1	MBUS：11 passos

Operação:

A Instrução MBUS pode habilitar comunicação com o mestre.

A string de comunicação envia um código HEX, incluindo código de comando, código de função e dados de comunicação. A instrução MBUS enviará o comando transferido de código ASCII para BUFF. O comando é uma string de comunicação composta por certos modos, tais como o modo RTU junto com código de verificação CRC (2 bytes) e caractere final (0DH+0AH).

String de recebimento inclui endereço, código da função e dados de comunicação. O caractere de início, de fim e código de verificação não serão salvos.

☑ O frame de comunicação da porta RS485 pode ser endereçado para um registro especial D8120. O CLP não aceitará os dados modificados em D8120 durante a operação MBUS.

☑ O frame de comunicação da placa opcional de expansão RS485/ RS232 pode ser ajustado pelo registro especial D8320. CLP não aceitará os dados modificados em D8320 durante operação MBUS.

☑ O comprimento de recebimento de dados 'm' deve ser ajustado para K0 quando nenhum dado for enviado.

☑ O programa pode aplicar muitas instruções como RS, MBUS, DTLK e RMIO, porém, deve ser assegurado que somente uma porta de comunicação será habilitada por vez. O tempo de chaveamento entre uma instrução e outra não deve ser menor que o tempo de uma varredura.

Especificação de comunicação:

<formato de comunicação [D8120], [D8320]>

D8120, D8320 são principalmente utilizados com a instrução F87 (MBUS). Eles também podem ser utilizados como um registro especial para outras instruções.

Porém, quando F87 (MBUS) é utilizado no programa, a parametrização de D8120, D8320 em relação a outras instruções de comunicação ou outros serão desabilitados. Favor parametrizar D8120, D8320 de acordo com as seguintes instruções.

Bit	Descrição	Conteúdo	
		0 (OFF)	1 (ON)
B0	Data bits	7 bit	8 bit
B1 B2	Paridade	B2,B1 (0,0): nenhum (0,1): IMPAR (1,0): PAR	
B3	Stop bits	1 bit	2 bit
B4 B5 B6 B7	Baud rate (bps)	B7,B6,B5,B4 (0,1,1,1):9.600 (1,0,0,0):19.200 (1,0,0,1):38.400 (1,0,1,0):57.600 (1,0,1,1):76.800	B7,B6,B5,B4 (1,1,0,0):128.000 (1,1,0,1):153.600 (1,1,1,0):307.200
B8～B12 *1	Reservado		
B13	Modo Modbus	(0) : Modo RTU	(1) : Modo ASCII
B14～B15*1	Reservado		

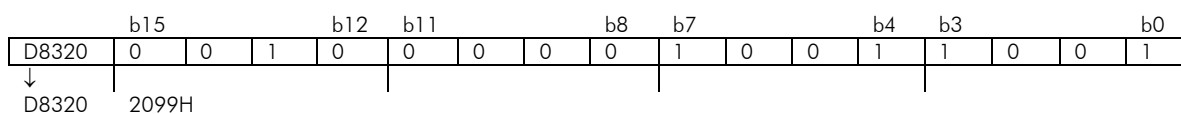
*1:B8～B12, B14, B15 é particularmente para outro modo de instrução. Quando na instrução F87 (MBUS), todos esses devem ser parametrizados com o valor 0.

☑ Exemplo de um telegrama de comunicação.

Favor parametrizar D8320 de acordo com os seguintes passos ou frame de comunicação periférico.

	b15			b12	b11			b8	b7			b4	b3			b0
D8320	0	0	1	0	0	0	0	0	1	0	0	1	1	0	0	1

↓

D8320 2099H

O marcador e registrador especial relacionado à instrução:

a) Para porta RS485

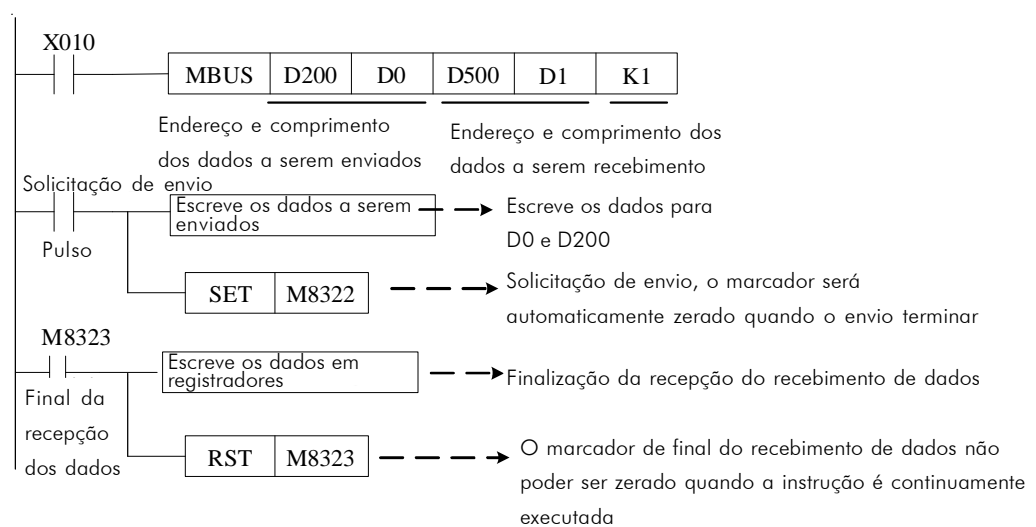
9) Pronto para enviar (M8121): o marcador será setado quando o CLP estiver pronto para receber dado. O marcador será automaticamente zerado quando iniciar a transmissão de dados.

10) Solicitação de envio (M8122): Quando M8122 for setado pelo pulso de inicio e fim de transmissão, a string de dados, definida no endereço (S) e de comprimento “m” será enviada. M8122 irá automaticamente zerar ao final da transmissão.

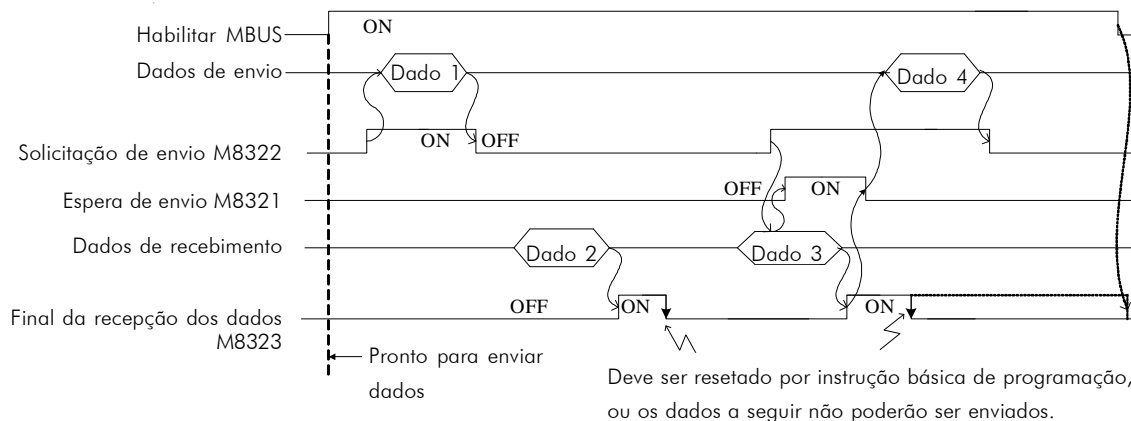
11) Final de envio: (M8123): M8123 será setado quando o envio estiver acabado. Favor zerar M8123 somente depois que os dados recebidos estiverem salvos em certos registros.
12) Marcador de erro (M8124): erro no recebimento (Modo RTU: erro no CRC; modo ASCII: erro no LRC ou erro no caractere final).
13) Falha na transmissão (M8129): Se nova tentativa de recebimento de dados não iniciar dentro do tempo especificado, o marcador de falha na transmissão será setado. Quando o envio terminar, M8123 será zerado e M8129 irá automaticamente resetado.
14) Frame de comunicação (D8120): refere-se ao frame da instrução MBUS.
15) Numero de dados restantes a serem enviados (D8122)
16) Numero de dados recebidos (D8123)
17) Tempo de watchdog para informar falha na transmissão (D8129): tempo de watchdog para falha de comunicação (5~255) x 10ms

b) Para placa de expansão RS485/ RS232
8) Pronto para enviar (M8321)
9) Solicitação de envio (M8322)
10) Final de envio (M8323)
11) Marcador de erro (M8124)
12) Falha na transmissão (M8329)
13) Frame de comunicação (D8320)
14) Numero de dados restantes a serem enviados (D8322)
15) Número de dados recebidos (D8323)
16) Tempo de watchdog (D8329)

Seqüência para enviar e receber dados

A instrução MBUS especifica o endereço do início dos dados e o número de dados do CLP, também define o primeiro endereço dos dados recebidos e número máximo de dados a receber. A seqüência para o envio e recebimento de dados MBUS é o seguinte: (Aplica-se placa de expansão RS485.)

Definição da porta de comunicação

Escreva os dados recebidos numa área de memória pré-definida

Enviar os dados recebidos para registro definido

Solicitação de envio M8322

☑ Quando X010 estiver energizado, a instrução MBUS será habilitada e o CLP estará pronto para receber dado.

☑ M8322 será setado por um pulso como em 'espera pelo recebimento de dados' ou em 'Recebendo dados'. O CLP enviará os dados iniciando com D200 e comprimento de dados D0 para fora. M8322 será zerado quando o envio terminar.

Final de recebimento dos dados M8323

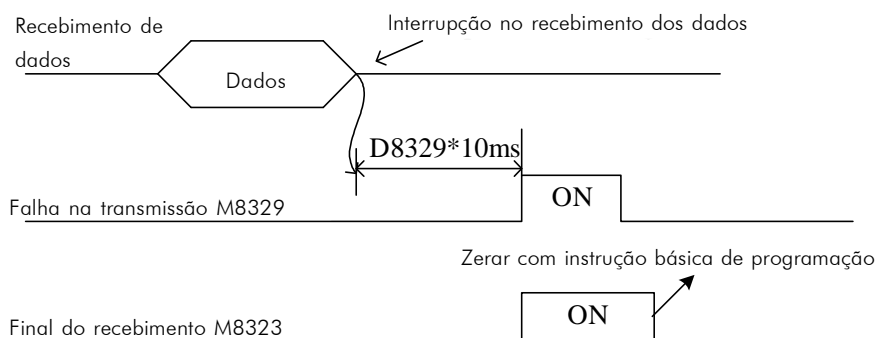
☑ Quando o marcador do final de recebimento dos dados M8323 for setado, o CLP irá armazenar todos os dados recebidos nos registradores correspondentes, então o marcador M8323 será zerado.

☑ Enquanto M8323 é zerado, o CLP estará pronto para receber dado. Se X010 estiver setado, a instrução MBUS será habilitada. Tal progresso será executado repetidamente.

☑ Quando (D1) = 0, a instrução MBUS é habilitada, M8323 não operará. Então, o CLP não entrará na próxima seqüência de recebimento de dados. Se D1=1, setando e resetando o marcador M8323, o CLP habilitará a próxima seqüência de recebimento de dados.

Falha na transmissão M8329

☑ Se houver interrupção no recebimento dos dados e o tempo definido no registrador D 8329 for atingido, o marcador M8329 será setado e a recepção de dados será interrompida. M8329 será automaticamente zerado enquanto M8323 desabilita o programa.

Recebimento de Dados (código ASCII) sem o caractere final também estão disponíveis com esta função.

Tempo de watchdog na transmissão

☑ Ajuste o tempo de watchdog para monitoração de falha de transmissão.
O tempo = valor ajustado X10ms, valor aceitável é 5~255. Caso o valor digitado esteja fora desta faixa, o registrador D8329 assumirá o valor 50ms.
Exemplo: tempo de falha de transmissão é ajustado em 50ms.

< Transmissão de dados de 16-bit> quando M8161=estiver desabilitado, (M8161 é um marcador especial usado pelas intruções RS, ASCI, HEX, CCD)

Os 16 bits de comunicação serão divididos em dois dados de 8 bits. Um é composto pelos 8 bits mais significativos e o outro é pelos oito menos significativos.

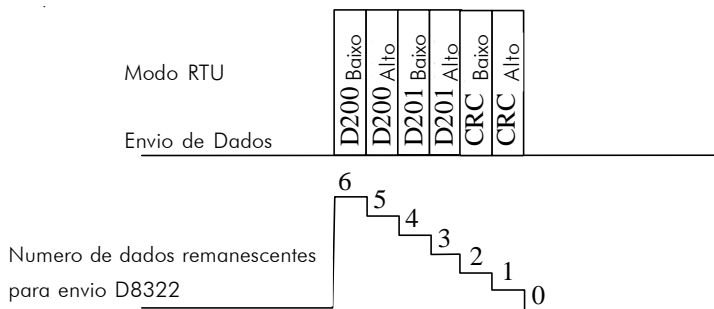
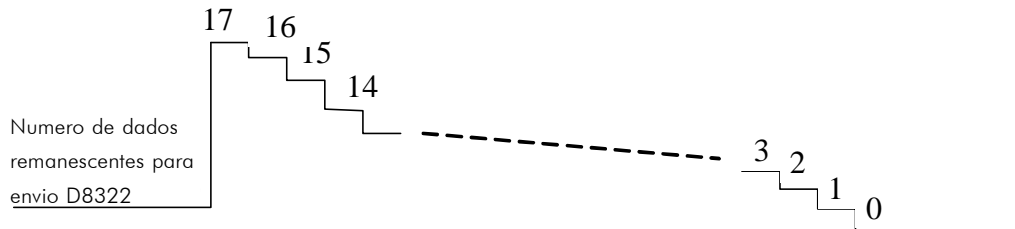
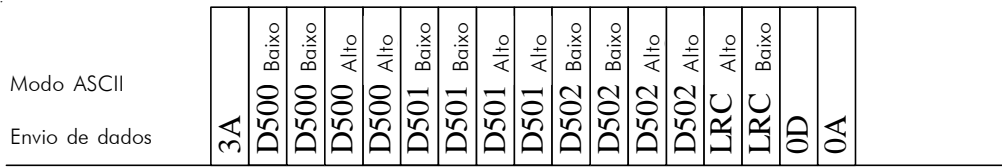
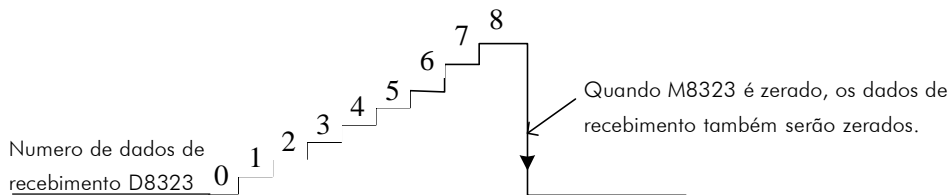
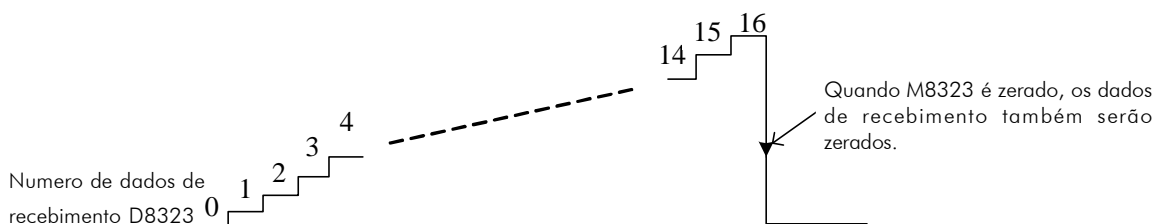
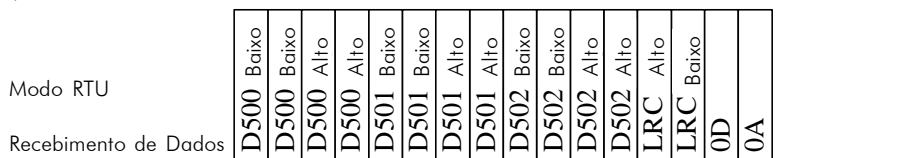
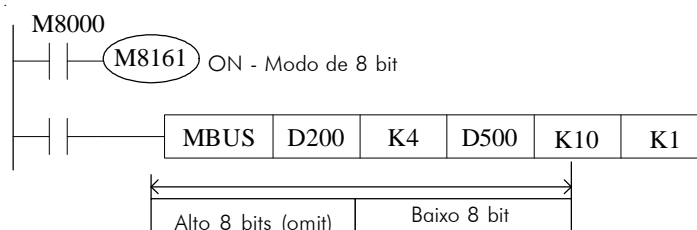
Dados de envio diferem nos modos ASCII e RTU

STX	D200 baixo	D200 alto	D201 baixo	D201 alto	Código de verificação	ETX
Caractere de início (3A)					LRC (ASCII)	Caractere final (0D0A)
Modo RTU (no)	↑ S. especifica o endereço inicial M especifica o número de bytes a serem enviados				CRC (RTU)	Modo RTU (no)

Recebimento de Dados

STX	D500 baixo	D500 alto	D501 baixo	D501 alto	D502 baixo	D502 alto	Código de verificação	ETX
Caractere de início (3A)							LRC (ASCII)	Caractere final (0D0A)
RTU (no)	↑ D. especifica o endereço de inicial Menor que n, ponto do limite superior para os dados de recebimento. Caractere de final EXT, ou n indicará fim de recebimento.						CRC (RTU)	

(1)Dados de envio e dados remanescentes para envio

Modo RTU

Modo ASCII

(1)Dados de envio e número de dados para envio
(2)Modo RTU

Modo ASCII

<Transmissão de dados de 8 bits (função de expansão)> M8161= setado (M8161 é um marcador especial utilizado pelas instruções RS,ASCI,HEX,CCD)

Somente os 8 bits mais baixos são significantes

Envio de dados difere nos modos ASCII e RTU

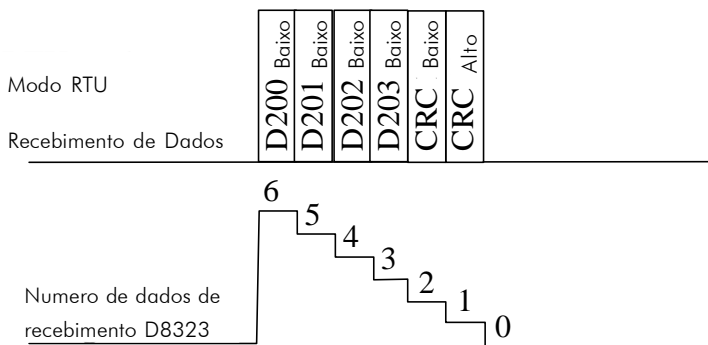
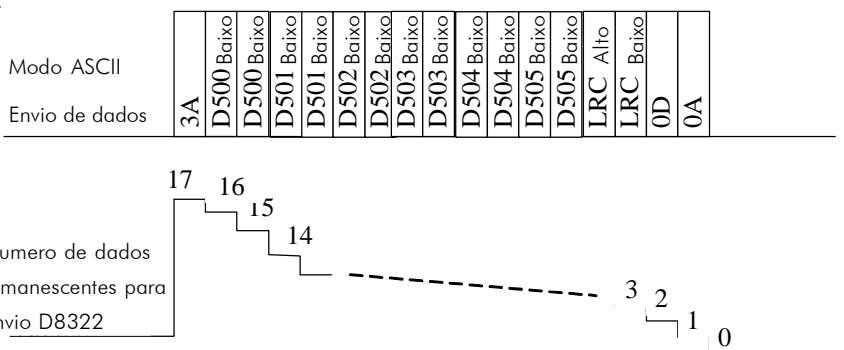
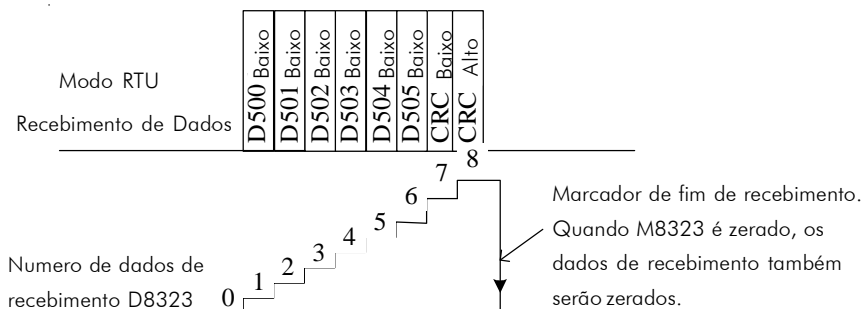
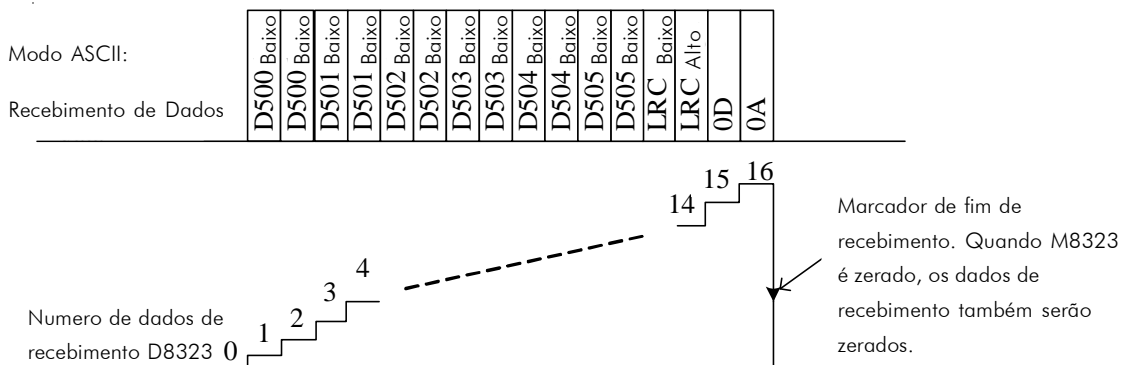
STX	D200 baixo	D201 baixo	D202 baixo	D203 baixo	Código de verificação	ETX
Caractere de início (3A)					LRC (ASCII)	Caractere final (0D0A)
Modo RTU (no)	↑ S. especifica o endereço inicial M especifica o número do byte de envio				CRC (RTU)	Modo RTU (no)

Recebimento de Dados

STX	D500 baixo	D501 baixo	D502 baixo	D503 baixo	D504 baixo	D505 baixo	Código de verificação	ETX
Caractere de início (3A)							LRC (ASCII)	Caractere final (0D0A)
RTU (no)	↑ D. especifica o endereço de início Menor que n, ponto do limite superior para os dados de recebimento. Caractere de final EXT, ou n indicará fim de recebimento.						CRC (RTU)	

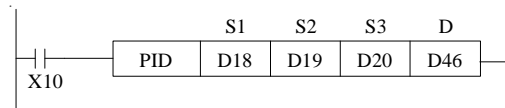
(1) Dados de envio e número de dados remanescentes

Modo RTU:

Modo RTU:

(1) Dados de recebimento e o número destes dados

Modo ASCII:

3.9.9 PID (FNC 88)

Mnemônico	Função	Operandos				Passos do programa
		S1	S2	S3	D	
PID FNC 88 (Módulo PID)	Recebe uma entrada de dados e calcula uma ação corretiva para um nível específico baseado em controle PID.	D☆		D☆ S3:S3~S3+6	D☆	PID: 9 passos

Operação:

Esta instrução pega uma variável atual (S2) e a compara com o valor de set-point definido em (S1). A diferença ou erro entre os dois valores é então processado através de uma malha PID para produzir um valor de correção para trazer o valor atual próximo ou igual ao valor de set-point. O PID calcula um fator de correção que é aplicado ao valor de saída atual e armazenado como um valor de saída corrigido no endereço de destino (D). Os parâmetros de regulagem da malha PID são armazenados em 25 registradores consecutivos S3+0 a S3+24.

Pontos para observar:

a) Toda aplicação PID é diferente. Haverá certa quantidade de "tentativa e erro" necessário para ajustar as variáveis em níveis ideais.

b) Uma característica de pré-ajuste está disponível e pode rapidamente providenciar valores iniciais para o processo PID.

c) Como 25 registros de dados são necessários para os parâmetros de regulagem da malha PID, o endereço inicial desta pilha de dados não poderá ser maior que D975. O conteúdo desta pilha de dados é explicado mais tarde nesta seção. Múltiplas instruções PID podem ser programadas, porém cada malha PID não deve ter registradores repetidos.

d) Existem limites de controle no CLP com a intenção de fazer com que as máquinas controladas por PID operem de maneira segura. Se for necessário zerar o Valor de set-point (S1) durante a operação, é recomendado desabilitar a malha PID e somente habilitar após entrar com o novo valor de set-point. Isto irá evitar que os limites de controle de segurança parem a operação da instrução PID de forma prematura.

e) A instrução PID tem um conjunto especial de códigos de erro a ela associados. Erros são identificados de maneira normal. Os códigos de erro associados a malha PID serão marcados por M8067 com o código de erro apropriado sendo armazenado em D8067. Estes endereços de erro não são exclusivos à instrução PID, então cuidado deve ser tomado para analisar corretamente os códigos de erro. Favor veja o capítulo 6, 'Endereços de Diagnóstico' para mais informações.

f) A malha PID não precisa ser executada com todos seus ganhos habilitados. Manipulando os parâmetros de regulagem P (proporcional), I (Integral) ou D (derivativo), controles podem ser acessados individualmente ou em um grupo definido/selecionado pelo usuário. Isto é detalhado mais tarde nesta seção.

Equações PID

Malha direta	$\Delta MV = Kp\{(EV_n - EV_{n-1}) + \frac{Ts}{T1} EV_n + D_n\}$ $EV_n = PV_{nf} - SV$ $Dn = \frac{T_D}{Ts + \alpha_D . T_D}(-2PV_{nf-1} + PV_{nf} + PV_{nf-2}) + \frac{\alpha_D . T_D}{Ts + \alpha_D . T_D} . D_{n-1}$ $MV_n = \sum \Delta MV$
Malha indireta	$\Delta MV = Kp\{(EV_n - EV_{n-1}) + \frac{Ts}{T1} EV_n + D_n\}$ $EV_n = SV - PV_{nf}$ $Dn = \frac{T_D}{Ts + \alpha_D . T_D}(2PV_{nf-1} - PV_{nf} - PV_{nf-2}) + \frac{\alpha_D . T_D}{Ts + \alpha_D . T_D} . D_{n-1}$ $MV_n = \sum \Delta MV \Delta$

PVnf = PVn + αPVnf-1 - PVn)

EVn = o Valor de Erro atual

EVn-1 = o Valor de Erro anterior

SV = o Valor de set-point (S1)

PVn = o Valor atual do Processo (S2)

PVnf = o Valor calculado do Processo

PVnf-1 = o Valor anterior do Processo

PVnf-2 = o penúltimo Valor de Processo

Δ MV = a mudança na Saída

KD = Constante de regulação do ganho derivativo

MVn = Valor atual da Saída de controle (D)

Dn = o Valor Derivativo

Dn-1 = o Valor Derivativo anterior

KP = Constante Proporcional

α=o Filtro de Entrada

TS = o Tempo de Amostragem

TI = Constante de Tempo da Integral

TD = Constante de Tempo Derivativo

Favor ver a seção de Parâmetros de regulagem para uma descrição mais detalhada dos parâmetros das variáveis e em que registro de memória devem ser ajustados.

Operação de Malha direta (S3+1, b0)

A operação direta é quando a atuação do valor de saída é diretamente refletida no valor atual do processo, ou seja, se o valor atual de processo estiver abaixo do set-point e a saída de correção aumentar, o valor atual deverá aumentar diretamente e proporcionalmente a correção aplicada.

A operação indireta é a condição quando para se reduzir o valor atual do processo, a saída de correção deverá aumentar seu valor e vice-versa.

Com controle PID supõe-se que algum trabalho necessita ser executado para trazer equilíbrio ao sistema. Então, ∆MV sempre terá um valor. Idealmente, um sistema estável necessitará de uma quantidade constante de trabalho para manter o set-point e o valor de processo iguais.

Parâmetros de regulagem da malha PID; S3

Os parâmetros de regulagem estão contidos em uma pilha de 25 registradores de dados. Alguns destes endereços requerem parametrização do usuário, alguns são reservados para operação interna e alguns retornam dados de saída da operação PID.

Parâmetros S3+0 a S3+6 devem ser parametrizados pelo usuário.

Parâmetro S3 + P	Nome/função do parâmetro	Descrição		Range ajustável
S3	Tempo de Amostragem (Ts)	O intervalo de tempo ajustado entre a leitura e o Valor de Processo atual do sistema (PVnf)		1~32767[ms]
S3+1	Ação – direção da reação e controle do alarme	BIT0	0:Operação direta 1: Operação indireta	Não aplicável
		BIT1	Valor de Processo (PVnf) habilitar alarme, OFF(0)/ON(1)	
		BIT2	Valor da Saída (MV) habilitar alarme, OFF(0)/ON(1)	
		BIT3-15	Reservado	
S3+2	Filtro de entrada (?)	Altera o efeito do filtro de entrada		0~99[%]
S3+3	Ganho proporcional (Kp)	Este é um fator utilizado para alinhar a saída proporcional em uma magnitude conhecida para a mudança no Valor de Processo (PVnf). Esta é a parte **P** da malha PID.		1~32767[%]
S3+4	Constante de tempo integral (TI)	Esta é a parte **I** da malha PID. Este é o tempo levado para que o valor de correção P seja totalmente aplicado na variável de saída. Selecionar 0 (zero) para este parâmetro desabilita o efeito **I**.		1~32767[x100 ms]
S3+5	Ganho derivativo (KD)	Este é um fator utilizado para alinhar a saída derivativa em uma proporção conhecida para uma mudança no Valor de Processo (PVnf).		0~100[%]
S3+6	Constante de tempo derivativo (TD)	Esta é a parte **D** da malha PID. Este é o tempo levado para que o valor derivativo corretivo alcance uma magnitude igual ao aplicado pelo proporcional ou parte **P** da malha. Selecionar 0 (zero) para este parâmetro desabilita o efeito **D**.		1~32767[x100 ms]
S3+7~S3 +19	Zerado para uso no processamento interno			
S3+20	Valor de Processo, valor máximo de variação	Ativo quando S3+1,b1 estiver setado. Este é um limite máximo para o Valor de Processo (PVnf) definido pelo usuário. Se o Valor de Processo (PVnf) exceder o limite, S3+24, bit b0 será setado.		0~32767
S3+21	Valor de Processo, valor mínimo de variação	Ativo quando S3+1,b1 estiver setado. Este é um limite mínimo para o Valor de Processo (PVnf) definido pelo usuário. Se o Valor de Processo (PVnf) exceder o limite, S3+24, bit b1 será setado.		
S3+22	Valor de Saída, Limite máximo de variação	Ativo quando S3+1,b2 estiver setado. Este é um limite máximo para a variação positiva que pode ocorrer em uma varredura PID. Se o Valor de Saída (MV) exceder isso, S3+24, bit b2 será setado.		
S3+23	Valor de Saída, Limite mínimo	Ativo quando S3+1,b2 estiver setado. Este é um limite máximo para a variação negativa que pode ocorrer em uma varredura PID. Se o Valor de Saída (MV) cair abaixo do limite mínimo, S3+24, bit b3 será setado.		
S3+24	Marcadores de alarme (Somente Leitura)	BIT0	Limite máximo excedido no Valor de Processo (PVnf)	Não aplicável
		BIT1	Abaixo do limite mínimo para o Valor de Processo (PVnf)	
		BIT2	Limite máximo de variação no Valor de Saída atingido (MV)	
		BIT3	Limite mínimo no Valor de Saída atingido (MV)	
		BIT4-15	Reservado	

Configurando da malha PID

A malha PID pode ser configurada para oferecer variações no controle PID. São as seguintes:

Método de controle	Seleção via registradores de regulagem			Descrição
	S3 +3 (KP)	S3+ 4 (TI)	S3 + 6 (TD)	
P	Valor do usuário	Ajustado a 0 (zero)	Ajustado a 0 (zero)	Somente efeito proporcional
PI	Valor do usuário	Valor do usuário	Ajustado a 0 (zero)	Efeito proporcional e integral
PD	Valor do usuário	Ajustado a 0 (zero)	Valor do usuário	Efeito proporcional e derivativo
PID	Valor do usuário	Valor do usuário	Valor do usuário	PID inteiro

Deve-se observar que em todas as circunstancias deverá haver um fator proporcional ou elemento 'P' na malha.

P – Ganho proporcional

Quando um fator proporcional é aplicado, a diferença entre o Valor Atual de Erro, EVn, e o Valor de Erro Anterior, EVn-1 é calculada. A correção proporcional é baseada na velocidade em que o Valor de Processo está se aproximando a (ou se distanciando de) um valor de set-point e NÃO na verdadeira distância entre PVnf e SV.

Nota: Outros sistemas PID poderão operar usando uma equação que calcula a mudança Proporcional baseada somente no tamanho do Valor de Erro Atual.

I – Ganho integral

Uma vez que uma correção proporcional foi aplicada ao uma situação de erro, o 'ajuste fino' da correção pode ser executado com o I ou elemento integral. Inicialmente somente uma mudança pequena é aplicada, mas conforme o tempo aumenta e o erro não é corrigido, o efeito integral é aumentado. É importante observar como TI realmente afeta a velocidade com qual a correção da integral total é aplicada.

Quanto menor for TI, maior será o efeito da integral.

Nota: O valor TI é ajustado no registrador de dados S3+4. Ajustar esta variável em zero desabilita o efeito integral.

D - Ganho derivativo

A função derivativa suplementa os efeitos causados pela resposta proporcional. O efeito derivativo é o resultado de um cálculo envolvendo elementos TD, TS, e o erro calculado. Isto faz com que a porção derivativa aplique inicialmente uma grande ação de saída que se dissipa rapidamente com o tempo. A velocidade desta dissipação pode ser controlada pelo valor TD: se o valor de TD for pequeno, então o efeito na aplicação do controle derivativo é aumentado.

O efeito inicial da derivativa pode ser relativamente severo e devido a isso, existe um efeito 'suavizador' que pode ser aplicado através do uso de KD, o ganho derivativo. A ação de KD pode ser considerada como um filtro, permitindo que a resposta derivativa seja escalada entre 0 e 100%.

Situações de "overshut" na regulagem, seja muito alta ou muito baixa, são na maioria das vezes associadas à porção Derivativa da equação por causa do grande fator de correção inicial.

Nota: O valor TD é ajustado no registro de Dados S3+6. Ajustar esta variável em zero desabilita o efeito Derivativo.

Uso do filtro de entrada αS3+2

Para prevenir a reação imediata e desenfreada da instrução PID a qualquer erro no Valor Atual, existe um mecanismo de filtragem que permite que a instrução PID observe e compute qualquer flutuação significativa durante três amostras.

O efeito quantitativo do filtro de entrada é o calculo de um Valor de Entrada filtrado para a instrução PID, tirada de uma porcentagem definida do Valor Atual e os dois Valores de Entrada filtrados anteriores.

Este tipo de filtragem é freqüentemente chamado de filtro de primeira ordem. É particularmente útil para remover os efeitos de ruídos de alta freqüência que possam aparecer em sinais de entrada recebidos dos sensores.

Quanto mais alto for ajustado o percentual de filtragem, mais longo o tempo de atraso na correção. Quando o filtro de entrada é ajustado a zero, isto efetivamente remove toda filtragem e permite que o Valor Atual seja utilizado diretamente como um Valor de Entrada.

Valores iniciais para malha PID

A instrução PID tem muitas funções que podem ser ajustadas e configuradas às necessidades do usuário. A dificuldade é achar um ponto de ajuste por onde iniciar o ajuste fino da malha PID as necessidades do sistema. As sugestões a seguir não serão ideais para todas as situações e aplicações, mas irão pelo menos dar aos usuários da instrução PID um ponto razoável por onde iniciar.

Um valor deve ser dado para todas as variáveis listadas a baixo antes de habilitar a instrução PID. Valores devem ser escolhidos para que o valor de saída não exceda ± 32767.

Ajustes iniciais recomendados:

TS = Deve ser igual ao ciclo de varredura total ou um múltiplo deste ciclo de varredura, por exemplo 2 vezes, 5 vezes, etc.

α= 50%

KP = Isto deve ser ajustado a um valor dependendo da ação corretiva máxima para alcançar o set-point – valores devem ser experimentados de um valor arbitrário de 75%

TI = Isto deveria idealmente ser 4 a 10 vezes maior que o tempo TD

KD = 50%

TD = Depende da resposta total do sistema, ou seja, não somente a velocidade em que o controlador programável reage, mas também qualquer válvula, bomba ou motor.

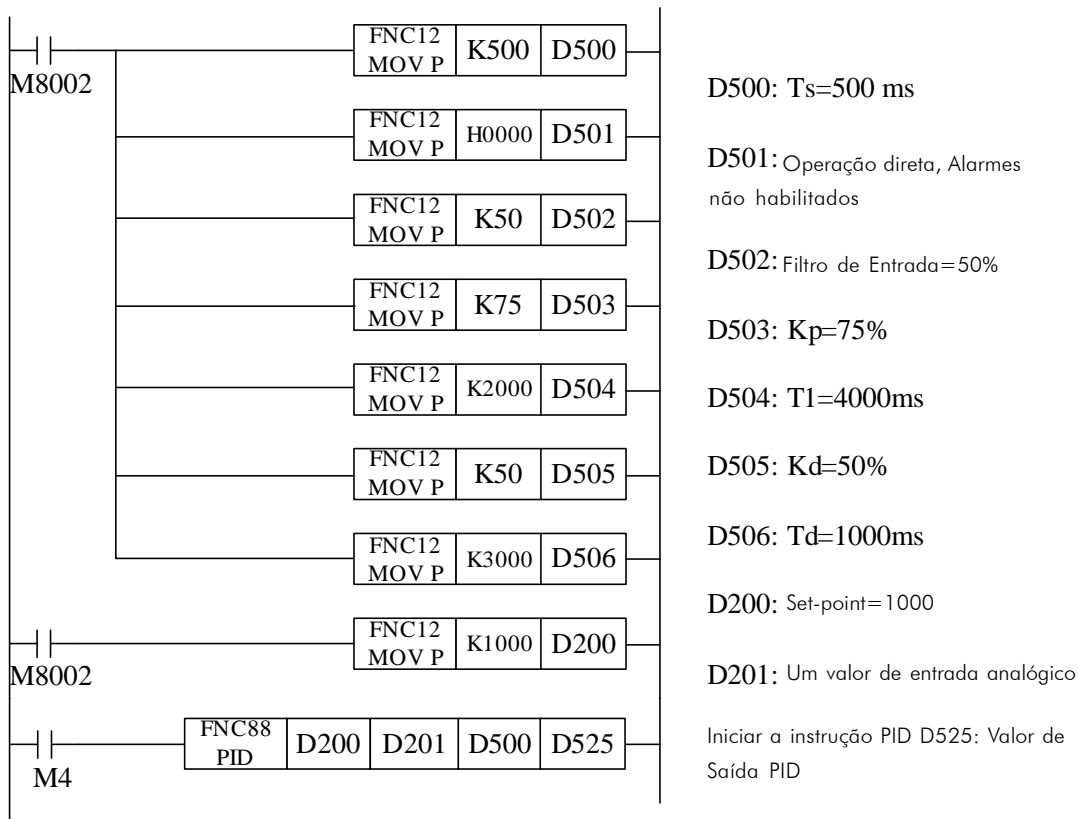
Para uma reação de sistema rápida TD será ajustado em um tempo rápido ou curto, porém nunca menor que TS. Um sistema de reação mais lento requer a duração mais longa do TD. Um valor inicial pode ser TD duas vezes o valor de TS.

Cuidado deve ser tomado quando ajustando variáveis PID para garantir a segurança do operador e evitar danos ao equipamento.

Quando utilizando TODOS os ganhos existentes na malha PID será necessário um certo grau de experimentação para ajustar a malha PID as condições que atendam as necessidades da aplicação. Uma maneira sensata de fazer isso é ajustar um parâmetro de cada vez usando porcentagens fixas, por exemplo, aumentando (ou diminuindo) o valor de KP de 10 em 10%. Selecionar programas PID sem a consideração devida resultará em um sistema mal configurado que não opera como necessário e causará frustração ao usuário. Favor lembrar que o processo PID é um cálculo puramente matemático e como tal não respeita a ‘qualidade’ dos dados variáveis fornecidos pelo usuário/sistema – o PID sempre processará sua função matemática PID com os dados disponíveis.

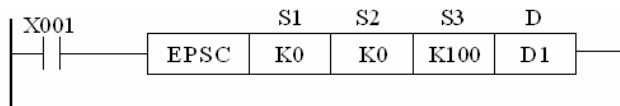
Exemplo de Ajustes PID

O programa parcial abaixo demonstra quais parâmetros devem ser ajustados para o funcionamento no TP-03. O primeiro passo ajusta os valores do usuário para S3+0 a S3+6. A instrução PID será ativada quando M4 for setado.

Da instrução PID no final do ladder, S1 = D200; S2 = D201; S3 = D500; e D ou M=D525

3.9.10 EPSC (FNC 89)

Mnemônico	Função	Operandos				Passos de Programa
		S1	S2	S3	D	
EPSC FNC 89	Leitura de entrada analógica do cartão de expansão TPW03-2AI	K, H (0~1)	K, H, KnX, KnY, KnM, KnS, T, C, D, V, Z		KnX, KnY, KnM, KnS, T, C, D, V, Z	EPSC, EPSCP: 10 passos

Operação:

O canal analógico do cartão de expansão TPW03-2AI, especificado em S1, terá seu valor convertido na faixa de valores entre S2 (valor mínimo) e S3 (valor máximo). O resultado desta conversão ficará salvo em D.

3.10 PONTO FLUTUANTE 1 & 2 - FUNÇÕES 110 À 129

Conteúdo:

Ponto Flutuante 1

ECMP -	Comparação com Ponto Flutuante	FNC 110
EZCP -	Comparação de área com Ponto Flutuante	FNC 111
☐☐☐ -	Não disponível	FNC 112 à 117
EBCD -	Ponto Flutuante -> Notação científica	FNC 118
EBIN -	Notação científica -> Ponto Flutuante	FNC 119

Floating Point 2

EADD -	Adição com Ponto Flutuante	FNC 120
ESUB -	Subtração com Ponto Flutuante	FNC 121
EMUL -	Multiplicação com Ponto Flutuante	FNC 122
EDIV -	Divisão com Ponto Flutuante	FNC 123
☐☐☐ -	Não disponível	FNC 124 à 126
ESQR -	Raiz Quadrada com Ponto Flutuante	FNC 127
PPP -	Não disponível	FNC 128
INT -	Integral do Ponto Flutuante	FNC 129

Lista de símbolos:

D – Endereço de destino.

S – Endereço de origem.

m, n - Número de endereços ativos, bits ou uma constante operacional.

Sufixos numéricos adicionais serão anexados se houver mais de um operando com a mesma função, ex: D1, S3 ou para endereços de lista/tabelados D3+0, S+9 etc.

MSB – Bit mais significante, por vezes usado para indicar o sinal matemático de um número, ex: positivo = 0, e negativo = 1.

LSB – Bit menos significante.

Modificações de instrução:

☐☐☐- Instrução de 16 bits, onde ☐☐☐ identifica a instrução mnemônica.

☐☐☐P - Instrução de 16 bits habilitada por pulso (simples).

D☐☐☐ - Instrução de 32 bits.

D☐☐☐P - Instrução de 32 bits habilitada por pulso (simples).

★ - Uma instrução repetitiva que mudará o valor de destino a cada varredura menos quando for habilitada por um pulso.

☆ - Um operando que não pode ser indexado, ex: A adição de V ou Z ou é inválida ou não terá efeito para o valor do operando.

3.10.1 ECMP (FNC 110)

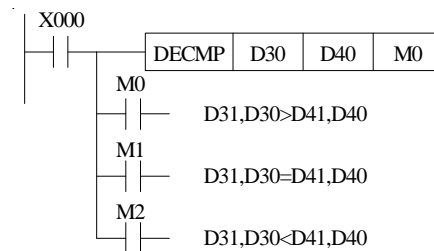
Mnemônico	Função	Operandos			Passos do programa
		S1	S2	D	
ECMP FNC 110 (Comparação com Ponto Flutuante)	Compara dois valores de ponto flutuante informando se eles são <, = ou >	K, H – valor da integral automaticamente convertida para ponto flutuante D – deve estar em formato de ponto flutuante (32bits)		Y, M, S Nota: 3 endereços consecutivos são utilizados.	DECMP, DECMPP: 13 passos

Operação:

Os dados de S1 são comparados aos dados de S2. O resultado é indicado por endereços de 3 dígitos especificados com o endereço inicial D. Os endereços de bit indicam:

S2 é menor que < S1 – endereço de bit D é setado.

S2 é igual a = S1 - endereço de bit D+1 é setado.

S2 é maior que > S1 - endereço de bit D+2 é setado.

Pontos para observar:

O estado dos endereços de destino será mantido mesmo que a instrução ECMP estiver desativada.

Comparações algébricas inteiras são utilizadas: ex: -1.79×10^{27} é menor que 9.43×10^{-15}

3.10.2 EZCP (FNC 111)

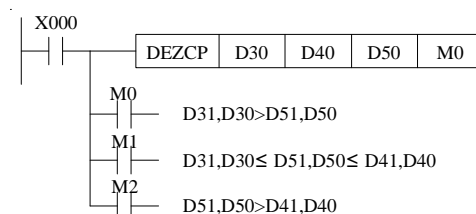
Mnemônico	Função	Operandos				Passos do programa
		S1	S2	S3	D	
EZCP FNC 111 (Compara uma área de valor em ponto flutuante)	Compara uma faixa de valores em ponto flutuante informando o resultado <, = ou >	K, H – valor da integral automaticamente convertida para ponto flutuante D – deve estar em formato de ponto flutuante (32 bits). Nota: S1 deve ser menor que S2			Y, M, S Nota: 3 endereços consecutivos são utilizados	DEZCP, DEZCPP: 13 passos

Operação:

A operação é igual a instrução ECMP, com excessão de que um valor único de dados (S3) é comparado a uma faixa de dados (S1 - S2).

S3 é menor que S1 e S2 – endereço de bit D é setado.

S3 está entre S1 e S2 - endereço de bit D+1 é setado.

S3 é maior que S2 - endereço de bit D+2 é setado.

3.10.3 EBCD (FNC 118)

Mnemônico	Função	Operandos		Passos do programa
		S	D	
EBCD FNC 118 (Conversão de ponto flutuante para notação científica)	Converte o formato ponto flutuante para o formato de número cientifico	D – deve estar em formato ponto flutuante (32 bits).	D - 2 endereços consecutivos são utilizados D - mantissa D+1 - expoente.	DEBCD, DEBCDP: 9 passos

Operação:

Converte um valor de ponto flutuante, em S, em mantissas separadas e partes exponenciais em D e D+1 (formato cientifico).

Pontos para observar:

a) A instrução deve estar em formato de doublé word. Os destinos D e D+1 representam a mantissa e o expoente do número em ponto flutuante respectivamente.

b) Para fornecer precisão máxima na conversão, a mantissa D estará na faixa 000 à 9999 (ou 0) e o expoente D+1 corrigido para um valor apropriado.

c) E.g. S= 3.4567 x 10^{-5} se tornará D= 34567, D+1 = -8

3.10.4 EBIN (FNC 119)

Mnemônico	Função	Operandos		Passos do programa
		S	D	
EBIN FNC 119 (Conversão de notação cientifica para ponto flutuante)	Converte um número em formato de notação cientifica ponto flutuante	D - 2 endereços consecutivos são utilizados S - mantissa S+1 - expoente	D – um valor de ponto flutuante (32 bits).	DEBIN, DEBINP: 9 passos

Operação:

Gera um número de ponto flutuante a partir de um de formato cientifico na origem S.

Pontos para observar:

a) A instrução deve estar em formato de double word. Os dados de origem S e S+1 representam a mantissa e o expoente do número ponto flutuante a ser gerado.

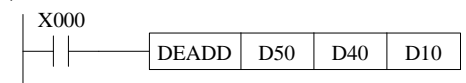
b) Para fornecer precisão máxima na conversão, a mantissa S estará no range 000 à 9999 (ou 0) e o expoente S+1 corrigido para um valor apropriado.

c) Ex: S= 5432, S+1 = 12 tornará D= 5,432 x 10^{9}

3.10.5 EADD (FNC 120)

Mnemônico	Função	Operandos			Passos do programa
		S1	S2	D	
EADD FNC 120 (Adição com ponto flutuante)	Adiciona dois números ponto flutuante	K, H – valor da integral automaticamente convertido para ponto flutuante D – deve estar em formato de ponto flutuante (32 bits).		D – um valor de ponto flutuante (32 bits).	DEADD, DEADDP: 13 passos

Operação:

Os valores de ponto flutuante são armazenados nos endereços de origem S1 e S2 são adicionados algebricamente e o resultado é armazenado no endereço de destino D.

Pontos para observar:

a) A instrução deve utilizar o formato de double word, ou seja, DEADD ou DEADDP. Todos os dados de origem e de destino serão double word, em outras palavras, utilizarão dois registros de dados consecutivos para armazenar dados (32 bits).
Com exceção de K ou H, todos os dados de origem serão considerados em formato ponto flutuante e o resultado armazenado no destino também será no mesmo formato.

b) Se um constante K ou H for usado como dado de origem, o valor é convertido para ponto flutuante antes da operação de adição.

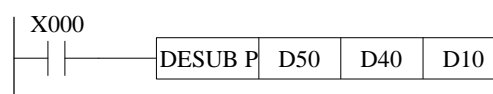
c) A adição é matematicamente correta. Ex.: 2,3456 x 10^{2} + (-5,6 x 10^{-1}) = 2,34 x 10^{2}

d) O mesmo endereço pode ser usado como origem e destino. Se este for o caso, então em operação contínua da instrução DEADD, o resultado da operação anterior será utilizado como novo valor de origem e um novo resultado será calculado.
Isto irá acontecer a cada varredura do programa a não ser que uma habilitação por pulso ou um intertravamento no programa seja utilizado.

e) Se o resultado do cálculo for zero "0", o marcador M8020 será setado.
Se o resultado do cálculo for maior que o maior número de em ponto flutuante, o marcador de carry, M8021 será setado e o resultado é ajustado para o maior valor.
Se o resultado do cálculo for menor que o menor número em ponto flutuante, o marcador borrow, M8022 será setado e o resultado é ajustado para o menor valor.

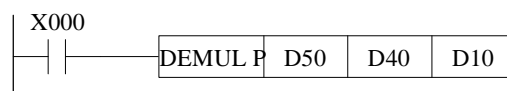
3.10.6 ESUB (FNC 121)

Mnemônico	Função	Operandos			Passos do programa
		S1	S2	D	
ESUB FNC 121 (Subtração com ponto flutuante)	Subtrai dois números no formato ponto flutuante	K, H – valor da integral automaticamente convertida para ponto flutuante. D – deve estar em formato de número ponto flutuante (32 bits).		D – um valor de ponto flutuante (32 bits).	DESUB, DESUBP: 13 passos

Operação:
O valor em ponto flutuante em S2 é subtraído do valor em ponto flutuante em S1 e o resultado é armazenado no endereço de destino D.

Pontos para observar:
Todos os pontos da instrução EADD se aplicam exceto que uma subtração é executada.

3.10.7 EMUL (FNC 122)

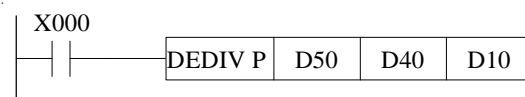
Mnemônico	Função	Operandos			Passos do programa
		S1	S2	D	
EMUL FNC 122 (Multiplicação com ponto flutuante)	Multiplica dois números ponto flutuante	K, H – valor da integral automaticamente convertido para ponto flutuante D – deve estar em formato ponto flutuante (32 bits).		D – um valor de ponto flutuante (32 bits).	DEMUL, DEMULP: 13 passos

Operação:
O valor de S1 é multiplicado com o valor de S2.
O resultado é armazenado em D.

Pontos para observar:
Pontos a, b, c e d da instrução EADD se aplicam exceto que uma multiplicação é executada.

3.10.8 EDIV (FNC 123)

Mnemônico	Função	Operandos			Passos do programa
		S1	S2	D	
EDIV FNC 123 (Divisão com ponto flutuante)	Divide dois número no formato ponto flutuante	K, H – valor da integral automaticamente convertida para ponto flutuante D – deve estar em formato ponto flutuante (32 bits).		D – um valor de ponto flutuante (32 bits).	DEDIV, DEDIVP: 13 passos

Operação:

O valor de S1 é dividido pelo valor de S2. O resultado da divisão é armazenado em D. O resto não é calculado.

Pontos para observar:

Pontos a, b, c e d da instrução EADD se aplicam exceto que uma divisão é executada.

☑ Se houver uma divisão por 0 (zero) então um erro de 'divisão por zero' acontecerá e a operação apresentará uma falha.

3.10.9 ESQR (FNC 127)

Mnemônico	Função	Operandos		Passos do programa
		S	D	
ESQR FNC 127 (Raiz Quadrada de ponto flutuante)	Calcula a raiz quadrada de um valor em ponto flutuante.	K, H – valor da integral automaticamente convertida para ponto flutuante. D – deve estar em formato de ponto flutuante (32 bits).	D – um valor de ponto flutuante (32 bits).	DESQR, DESQRP: 9 passos

Operação:

Uma raiz quadrada é executada com um valor no formato de ponto flutuante S e o resultado é armazenado em D.

Pontos para observar:

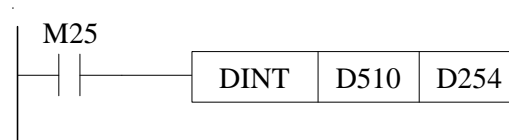
Pontos a, b, c e d da instrução EADD se aplicam exceto que uma raiz quadrada é executada.

☑ Se S for negativo então um erro acontecerá e o marcador de erro M8067 é setado.

3.10.10 INT (FNC 129)

Mnemônico	Função	Operandos		Passos do programa
		S	D	
INT FNC 129 (Conversão de ponto flutuante para ponto fixo)	Converte um número de formato de ponto flutuante para ponto fixo	K, H – valor da integral automaticamente convertido para ponto flutuante. D – deve estar em formato de ponto flutuante (32 bits).	D – formato ponto fixo para INT, INTP - 16 bits para DINT, DINTP – 32 bits	INT, INTP: 5 passos DINT, DINTP: 9 passos

Operação:

O valor em ponto flutuante em S é arredondado para baixo até o valor interior mais próximo e normalmente armazenado em formato binário em D.

Pontos para observar:

a) Os dados de origem são sempre uma double word (32 bits); um valor de ponto flutuante.
Para operação com word (16 bits) o destino é um valor de 16 bits.
Pra operação com doublé word (32 bit) o destino é um valor de 32 bits.

b) Esta instrução é o inverso da instrução FLT.

c) Se o resultado for 0, então o marcador M8020 é setado.
Se o dado de origem não for um número inteiro, deve ser arredondado para baixo. Neste caso o marcador borrow M8021 é setado para indicar um valor arredondado.
Se o número inteiro estiver fora da faixa válida para do endereço de destino, então o bit de carry M8022 é setado.

Note: Se o bit M8022 for setado, o valor no endereço de destino não será válido.

3.11 TRIGONOMETRIA - FNC 130 À FNC 139

Conteúdo:
Ponto Flutuante 3

SIN -	Seno	FNC 130
COS -	Co-seno	FNC 131
TAN -	Tangente	FNC 132
ASIN -	ARC Seno	FNC 133
ACOS -	ARC Co-seno	FNC 134
ATAN -	ARC Tangente	FNC 135
RAD -	Grau ao Radiano	FNC 136
DEG -	Radiano ao Grau	FNC 137
☐☐☐-	Não Disponível	FNC 138 à 139

Lista de símbolos:
D – Endereço de destino.
S – Endereço de origem.
m, n - Número de endereços ativos, bits ou uma constante operacional.
Sufixos numéricos adicionais serão anexados se houver mais de um operando com a mesma função, ex: D1, S3 ou para endereços de lista/tabelados D3+0, S+9 etc.
MSB – Bit mais significante, por vezes usado para indicar o sinal matemático de um número, ex: positivo = 0, e negativo = 1.
LSB – Bit menos significante.

Modificações de instrução:
☐☐☐- Instrução de 16 bits, onde ☐☐☐ identifica a instrução mnemônica.
☐☐☐P - Instrução de 16 bits habilitada por pulso (simples).
D☐☐☐ - Instrução de 32 bits.
D☐☐☐P - Instrução de 32 bits habilitada por pulso (simples).

★ - Uma instrução repetitiva que mudará o valor de destino a cada varredura menos quando for habilitada por um pulso.
☆ - Um operando que não pode ser indexado, ex: A adição de V ou Z ou é inválida ou não terá efeito para o valor do operando.

3.11.1 SIN (FNC 130)

Mnemônico	Função	Operandos		Passos do programa
		S	D	
SIN FNC 130 (Seno)	Calcula o seno de um número de ponto flutuante	D – deve estar em formato de número ponto flutuante (32 bits).(radianos)	D – um valor em ponto flutuante (32 bits).	DSIN, DSINP: 9 passos

Conteúdo:

Esta instrução executa a operação matemática SENO em ponto flutuante do valor em S. O resultado é armazenado em D.

Pontos para observar:

a) A instrução deve usar o formato de double word, ou seja, DSIN ou DSINP. Todos os dados de origem e destino serão double word, em outras palavras, utilizarão dois registros de dados consecutivos para armazenar os dados (32 bits).

Os dados de origem são considerados em formato de ponto flutuante e o destino também está no mesmo formato.

Ângulos dos Radianos

Abaixo está um exemplo de programa para o cálculo de ângulos em radianos usando ponto flutuante.

K45 graus para D0

K90 graus para D0

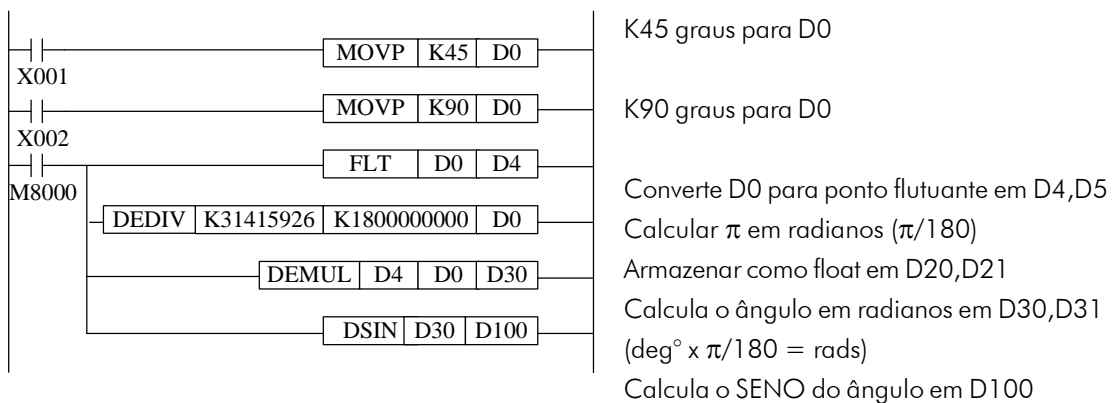
Converte D0 para ponto flutuante em D4,D5

Calcular π em radianos (π/180)

Armazenar como float em D20,D21

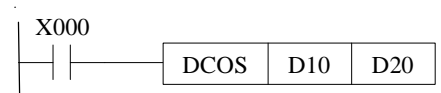
Calcula o ângulo em radianos em D30,D31

(deg° x π/180 = rads)

Calcula o SENO do ângulo em D100

3.11.2 COS (FNC 131)

Mnemônico	Função	Operandos		Passos do programa
		S	D	
COS FNC 131 (Co-seno)	Calcula o co-seno do valor em ponto flutuante	D – deve estar em formato ponto flutuante (32 bits).	D – Valor em ponto flutuante (32 bits).	DCOS, DCOSP: 9 passos

Conteúdo:

Esta instrução executa a operação matemática CO-SENO de um valor em ponto flutuante em S. O resultado é armazenado em D.

Pontos para observar:

Todos os pontos para a instrução SIN se aplicam menos que o COS é calculado.

3.11.3 TAN (FNC 132)

Mnemônico	Função	Operandos		Passos do programa
		S	D	
TAN FNC132 (Tangente)	Calcula o tangente de um valor em ponto flutuante	D – deve estar em formato ponto flutuante (32 bits).	D – valor em ponto flutuante (32 bits).	DTAN, DTANP: 9 passos

Conteúdo:

Esta instrução executa a operação matemática TANGENTE de um valor em ponto flutuante em S. O resultado é armazenado em D.

Pontos para observar:

Todos os pontos para a instrução SIN se aplicam menos que o COS é calculado.

3.11.4 ASIN (FNC 133)

Mnemônico	Função	Operandos		Passos do programa
		S	D	
ASIN FNC133 (ARCO SENO)	Calcula o arco seno de um valor em ponto flutuante	D -1≤S<1	D	DASIN, DASINP: 9 passos

Conteúdo:

Esta instrução calcula o ARC SIN (função inversa de SIN) os dados em S, então enviam o resultado para D.

Exemplo:

(D11, D10) RAD (valor em ponto flutuante)

↓

(D21, D20) ASIN (valor em ponto flutuante)

3.11.5 ACOS (FNC 134)

Mnemônico	Função	Operandos		Passos do programa
		S	D	
ACOS FNC134	ARC COS de um valor em ponto flutuante	D -1≤S<1	D	DACOS, DACOSP: 9 passos

Conteúdo:

Esta instrução calcula o ARC COS (função inversa de COS) os dados em S, então enviam o resultado para D.

Exemplo:

(D11, D10) RAD (valor em ponto flutuante)

↓

(D21, D20) ACOS resultado (valor em ponto flutuante)

3.11.6 ATAN (FNC 135)

Mnemônico	Função	Operandos		Passos do programa
		S	D	
ATAN FNC135	ARC TAN de um valor em ponto flutuante	D -π/2~π/2	D	DATAN, DATANP: 9 passos

Conteúdo:

Esta instrução calcula o ARC TAN (função inversa de TAN) os dados em S, então enviam o resultado para D.

Exemplo:

(D11, D10) RAD (valor em ponto flutuante)

↓

(D21, D20) ATAN resultado (valor em ponto flutuante)

3.11.7 RAD (FNC 136)

Mnemônico	Função	Operandos		Passos do programa
		S	D	
RAD FNC136	Converte um valor de graus para radianos	S – Valor em ponto flutuante	D – Valor em ponto flutuante	DRAD DRADP: 9 passos

Conteúdo:

Esta instrução converte unidades de ângulos para radianos.

3.11.8 DEG (FNC 137)

Mnemônico	Função	Operandos		Passos do programa
		S	D	
DEG FNC137	Converte um valor de radianos para graus	S – um úmero no formato de ponto flutuante (32 bits).	D - um número no formato de ponto flutuante (32 bits).	DDEG DDEGP: 9 passos

3.12 OPERAÇÕES DE DADOS 2 - FNC 140 À FNC 149

Conteúdo:

☐☐☐ -	Não Disponível	FNC 140 à 146
SWAP -	SWAP	FNC 147
☐☐☐ -	Não Disponível	FNC 148 à 149

Lista de símbolos:
D – Endereço de destino.
S – Endereço de origem.
m, n - Número de endereços ativos, bits ou uma constante operacional.

Sufixos numéricos adicionais serão anexados se houver mais de um operando com a mesma função, ex: D1, S3 ou para endereços de lista/tabelados D3+0, S+9 etc.

MSB – Bit mais significante, por vezes usado para indicar o sinal matemático de um número, ex: positivo = 0, e negativo = 1.

LSB – Bit menos significante.

Modificações de instrução:
☐☐☐- Instrução de 16 bits, onde ☐☐☐ identifica a instrução mnemônica.
☐☐☐P - Instrução de 16 bits habilitada por pulso (simples).
D☐☐☐ - Instrução de 32 bits.
D☐☐☐P - Instrução de 32 bits habilitada por pulso (simples).

★ - Uma instrução repetitiva que mudará o valor de destino a cada varredura menos quando for habilitada por um pulso.

☆ - Um operando que não pode ser indexado, ex: A adição de V ou Z ou é inválida ou não terá efeito para o valor do operando.

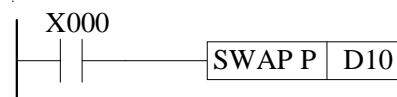
3.12.1 SWAP (FNC 147)

Mnemônico	Função	Operandos	Passos do programa
		S	
FNC 147 (SWAP) ★	O byte alto é trocado de posição com o byte baixo de um determinado endereço	KnY, KnM, KnS, T, C, D, V, Z	SWAP, SWAPP: 5 passos DSWAP, DSWAPP: 9 passos

Conteúdo:

O byte mais alto é trocado de posição com o conteúdo do byte baixo de um determinado endereço de word.

Esta instrução é equivalente à operação 2 do FNC 17 XCH

X000 — SWAP P | D10

Pontos para observar:

a) Para execução com word (16 bits) o byte mais alto e o mais baixo do endereço de origem são trocados.

b) Para execução com double word (32 bits) o byte mais alto e o mais baixo de cada um dos dois endereços de 16 bits serão trocados.

Resultado do DSWAP(P) D10:

Valores estão em Hex para facilitar visualização		Antes DSWAP	Depois DSWAP
D10	Byte 1	1FH	8BH
	Byte 2	8BH	1FH
D11	Byte 1	C4H	35H
	Byte 2	35H	C4H

c) Se a operação desta instrução for habilitada a cada ciclo de varredura, então o valor do endereço de origem irá voltar ao seu valor original a cada dois ciclos de varredura. O uso de uma habilitação por pulso ou um intertravamento no programa é recomendado.

3.13 INSTRUÇÃO DE POSICIONAMENTO – FNC 156 À FNC 159

Conteúdo:

□□□ -	Não Disponível	FNC 150 à 155
ZRN -	Retorno a posição Zero	FNC 156
PLSV -	saída do pulso de velocidade variável	FNC 157
DRVI -	posicionamento incremental	FNC 158
DRVA -	posicionamento absoluto	FNC 159

Lista de símbolos:

D – Endereço de destino.

S – Endereço de origem.

m, n - Número de endereços ativos, bits ou uma constante operacional.

Sufixos numéricos adicionais serão anexados se houver mais de um operando com a mesma função, ex: D1, S3 ou para endereços de lista/tabelados D3+0, S+9 etc.

MSB – Bit mais significante, por vezes usado para indicar o sinal matemático de um número, ex: positivo = 0, e negativo = 1.

LSB – Bit menos significante.

Modificações de instrução:

□□□- Instrução de 16 bits, onde □□□ identifica a instrução mnemônica.

□□□P - Instrução de 16 bits habilitada por pulso (simples).

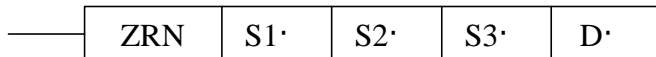
D□□□ - Instrução de 32 bits.

D□□□P - Instrução de 32 bits habilitada por pulso (simples).

★- Uma instrução repetitiva que mudará o valor de destino a cada varredura menos quando for habilitada por um pulso.

☆ - Um operando que não pode ser indexado, ex: A adição de V ou Z ou é inválida ou não terá efeito para o valor do operando.

3.13.1 ZRN (FNC 156)

Mnemônico	Função	Operandos				Passos do programa
		S1	S2	S3	D	
ZRN FNC 156	Volta ao ponto zero depois do LIGAR da máquina ou ajuste inicial.	K,H,KnY, KnM, KnS, T, C, D, V, Z		X,Y,M,S	Y	

☑ Para FNC158 (DRVI) e FNC159 (DRVA) , o CLP irá controlar a posição atual, aumentando ou diminuindo os pulsos para frente e para trás produzido pelo próprio CLP, ele armazenará os valores atualizados no registro (Y000: [D8141, D8140], Y00: [D8143,D8142]). Através estes valores, o CLP sempre saberá a posição da máquina. Porém, quando a força for desligada, os dados serão perdidos. Conseqüentemente, para resolver o problema, é necessário executar FNC156 (ZRN) quando a maquina for energizada ou editar o programa de forma a executar a função de retorno ao zero no início da operação.

a) Usuários podem especificar velocidade para executar a função de retorno à zero [S1] como, 16-bits 10 à 32,767Hz ou 32-bits 10 à 100kHz.

b) Usuários podem especificar a velocidade de desaceleração [S2] de 10 à 32,767Hz

c) Se qualquer outro endereço, fora um endereço de entrada (X), for especificado para ser o ponto de aproximação [S3], ele será afetado pelo ciclo operacional do CLP e o erro na precisão do ponto zero poderá ser grande.

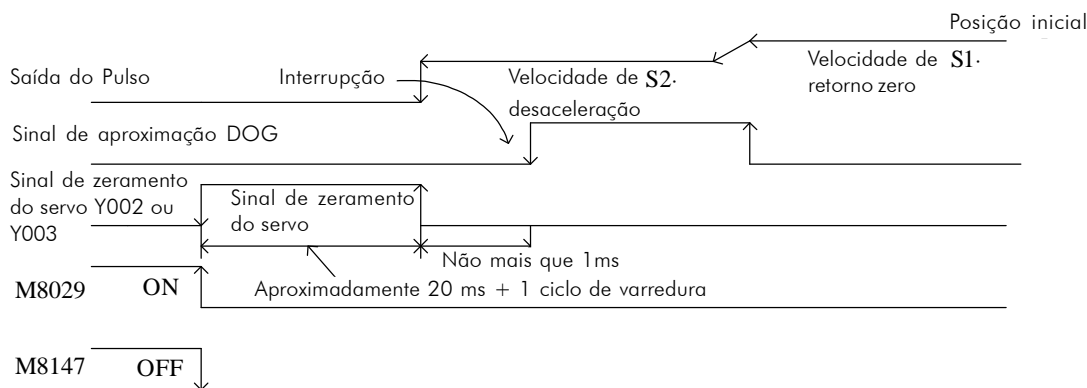
d) Somente Y000 ou Y001 pode ser usado para a saída do pulso [D].

Função de saída desta instrução:

Se M8140 estiver setado, o sinal de zerar será encaminhado para o servo motor quando a função de retorno ao zero estiver completa.

O sinal de zerar pode variar dependendo da saída de pulso a ser utilizada:

Saída do pulso[Y000]->sinal de zerar [Y002]

Saída do pulso[Y001]-> sinal de zerar [Y003]

☑ A seqüência de execução para esta instrução:

a) Conforme a instrução é habilitada, a máquina irá se mover na velocidade ajustada S1.

☑ No progresso de retorno a posição zero, a máquina irá parar quando o sinal de habilitação for zerado.

☑ Se o sinal de habilitação D estiver desenergizado e o endereço do monitor da saída de pulso estiver setado, a máquina não aceitará tal instrução.

b) Quando o sinal de aproximação (DOG) receber um flanco de descida, a máquina irá movimentar-se na velocidade de desaceleração S2.

c) Quando o sinal de aproximação (DOG) estiver desenergizado e a saída de pulso parar, o dado '0' será escrito para o registro atual (Y000: [D8141, D8140], Y001: [D8143, D8142]). Quando M8140 estiver setado, o CLP mandará um sinal de zerar. Depois de acabar o zeramento, o M8029 será setado, assim como o endereço de monitoração da saída de pulso (000: M8147; Y001: M8148) será desenergizado.

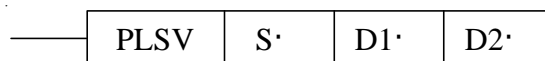
Número do endereço relacionado:
D8141 (dígito superior) & D8140 (dígito inferior): Registro do valor atual de Y000 (32-bit)
D8143 (dígito superior) & D8142 (dígito inferior): Registro do valor atual de Y001 (32-bit)
M8145: Y000 parada da saída do pulso (imediato)
M8146: Y001 parada da saída do pulso (imediato)
M8147: Y000 monitoração da saída do pulso (BUS/READY)
M8148: Y001 monitoração da saída do pulso (BUS/READY)

Consideração:

Caso a função de busca do zero não seja completada, inicie o processo de retorno ao zero do lado frontal do sinal de aproximação.

Ao encontrar o ponto zero, o valor atual é zerado no registro (Y000: [D8141, D8140], Y001: [D814, D8142]).
Observe com atenção a temporização da habilitação desta instrução.

3.13.2 PLSV (FNC 157)

Mnemônico	Função	Operandos			Passos do programa
		S	D1	D2	
PLSV FNC 157	Saída de pulso com velocidade variável	K,H,KnY, KnM, KnS, T, C, D, V, Z	Y	X,Y,M	

☑ Esta é uma instrução que gera pulsos numa determinada saída com velocidade variável, esta saída define a direção da rotação.

a) Usuários podem usar freqüências de pulso de saída [S1] de, 16-bit 1 à 32.767Hz/-1 à 32.767Hz ou 32-bit 1 à 100kHz/-1 à 100kHz.

b) Somente Y000 ou Y001 pode ser usado para a saída do pulso [D1].
Devido à natureza da saída de alta velocidade, unidades de saída do tipo transistor devem ser usadas com esta instrução. Saídas do tipo relé terão sua vida útil muito reduzida, e poderão ocasionar sinais falsos.

c) Saídas do sinal de direção da rotação [D2] operaram da seguinte maneira: se [D2] = OFF, rotação = negativa, se [D2] = ON, rotação = positiva.

☑ A freqüência do pulso [S] pode ser mudada mesmo quando pulsos estiverem sendo gerados.

☑ Aceleração/desaceleração não são executados no início/parada. Se for necessário iniciar ou parar o movimento suavemente, aumente ou diminua a freqüência do pulso de saída [S] usando instrução de RAMP FNC67.

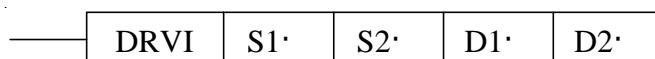
☑ Se o contato de habilitação da instrução desligar enquanto pulsos estiverem sendo gerador, a máquina irá parar imediatamente, mas não irá desacelerar até 0.

☑ Números de endereços relacionados.

D8141 (dígito superior) & D8140 (dígito inferior): Registro de valor atual de Y000 (32-bit)
D8143 (dígito superior) & D8142 (dígito inferior): Registro de valor atual de Y001 (32-bit)
M8145 : Y000 parar saída de pulso (imediato)
M8146 : Y001 parar saída de pulso (imediato)
M8147 : Y000 monitorar saída de pulso (BUS/READY)
M8148 : Y001 monitorar saída de pulso (BUS/READY)

Observe com atenção a temporização da habilitação desta instrução.

3.13.3 DRVI (FNC 158)

Mnemônico	Função	Operandos				Passos do programa
		S1	S2	D1	D2	
DRVI FNC 158	Posicionamento incremental	K,H,KnY, KnM, KnS, T, C, D, V, Z		Y	Y,M,S	

☑ Esta instrução é para realização de um posicionamento em velocidade única com movimentos incrementais.
a) O número máximo de pulsos [S1] disponíveis é: 16-bits -32.768 a 32.767 pulsos ou 32-bits. -2.147.483.648 a 2.147.483.648 pulsos.

b) Usuários podem usar as freqüências dos pulsos de saída [S2], 16-bits 10 a 32.767Hz ou 32-bits 10 a 100 kHz.

c) Somente Y000 ou Y001 pode ser usado para a saída do pulso [D1].
Devido à natureza da saída de alta velocidade, unidades de saída do tipo transistor devem ser usadas com esta instrução. Saídas do tipo relé terão sua vida útil muito reduzida, e poderão ocasionar sinais falsos.

d) Saídas do sinal de direção da rotação [D2] operaram da seguinte maneira: se [D2] = OFF, rotação = negativa, se [D2] = ON, rotação = positiva.

☑ Endereço Relacionado:
D8141 (dígito superior) & D8140 (dígito inferior): Registro de valor atual de Y000 (32-bit)
D8143 (dígito superior) & D8142 (dígito inferior): Registro de valor atual de Y001 (32-bit)
No sentido inverso, o valor atual em registro irá diminuir.

☑ Se o conteúdo de um operando for alterado enquanto a instrução é executada, não será refletido na operação. O novo conteúdo torna-se efetivo na próxima vez que a instrução for executada.

☑ Se o contato de habilitação da instrução desligar enquanto a instrução estiver sendo executada, a máquina desacelerará e ir parar. Nesta hora o marcador M8029 não será setado.

☑ Uma vez que o contato de habilitação da instrução estiver desenergizado, não será possível habilitar novamente a instrução enquanto o marcador de saída de pulso Y000: [M8147] Y001: [M8148] estiver setado.

☑ Para operação no método de inversor incremental, a distância do percurso da posição atual é especificada ou por uma posição ou um símbolo negativo.

☑ O tempo de aceleração e desaceleração é ajustado por D8148.

3.13.4 DRVA (FNC 159)

Mnemônico	Função	Operandos				Passos do programa
		S1	S2	D1	D2	
DRVA FNC 159	Posicionamento absoluto	K,H,KnY, KnM, KnS, T, C, D, V, Z		Y	Y,M,S	

☑ Esta instrução é para posicionamento em velocidade única usando um ponto zero e medidas absolutas.

a) A posição de destino para posicionamento absoluto [S1] pode ser: 16-bits -32.768 a 32.767 pulsos ou 32-bits -2.147.483.648 a +2.147.483.647 pulsos.

b) Usuários podem usar as freqüências dos pulsos de saída [S2], 16-bits 10 a 32.767Hz ou 32-bits 10 a 100 kHz.

c) Somente Y000 ou Y001 pode ser usado para a saída do pulso [D1].
Devido à natureza da saída de alta velocidade, unidades de saída do tipo transistor devem ser usadas com esta instrução. Saídas do tipo relé terão sua vida útil muito reduzida, e poderão ocasionar sinais falsos.

d) Saída do sinal de direção da rotação [D2] operaram da seguinte maneira: se [D2] = OFF, rotação = negativa, se [D2] = ON, rotação = positiva.

☑ Endereço Relacionado:
D8141 (dígito superior) & D8140 (dígito inferior): Registro de valor atual de Y000 (32-bit)
D8143 (dígito superior) & D8142 (dígito inferior): Registro de valor atual de Y001 (32-bit)
No sentido inverso, o valor atual em registro irá diminuir.

☑ Se o conteúdo de um operando for alterado enquanto a instrução é executada, não será refletido na operação. O novo conteúdo torna-se efetivo na próxima vez que a instrução for executada.

☑ Se o contato de habilitação da instrução desligar enquanto a instrução estiver sendo executada, a máquina desacelerará e irá parar. Nesta hora o marcador M8029 não será setado.

☑ Uma vez que o contato de habilitação da instrução estiver desligado, não será possível habilitar novamente a instrução enquanto o marcador de saída de pulso Y000: [M8147] Y001: [M8148] estiver setado.

☑ Para operação no método de inversor incremental, a distância do percurso da posição atual é especificada ou por uma posição ou um símbolo negativo.

☑ O tempo de aceleração e desaceleração é ajustado por D8148.

3.14 CONTROLE DO RELÓGIO DE TEMPO REAL 160 A 169

Conteúdo:

TCMP -	Compara Horário	FNC 160
TZCP -	Compara faixa de horário	FNC 161
TADD -	Soma Horário	FNC 162
TSUB -	Subtrai Horário	FNC 163
☐☐☐ -	Não disponível	FNC 164 à 165
TRD -	Ler dados RTC	FNC 166
TWR -	Ajustar dados RTC	FNC 167
☐☐☐ -	Não disponível	FNC 168 à 169

Lista de símbolos:

D – Endereço de destino.
S – Endereço de origem.
m, n - Número de endereços ativos, bits ou uma constante operacional.
Sufixos numéricos adicionais serão anexados se houver mais de um operando com a mesma função, ex: D1, S3 ou para endereços de lista/tabelados D3+0, S+9 etc.
MSB – Bit mais significante, por vezes usado para indicar o sinal matemático de um número, ex: positivo = 0, e negativo = 1.
LSB – Bit menos significante.

Modificações de instrução:
☐☐☐- Instrução de 16 bits, onde☐☐☐ identifica a instrução mnemônica.
☐☐☐P - Instrução de 16 bits habilitada por pulso (simples).
D☐☐☐ - Instrução de 32 bits.
D☐☐☐P - Instrução de 32 bits habilitada por pulso (simples).

★ - Uma instrução repetitiva que mudará o valor de destino a cada varredura menos quando for habilitada por um pulso.
☆ - Um operando que não pode ser indexado, ex: A adição de V ou Z ou é inválida ou não terá efeito para o valor do operando.

3.14.1 TCMP (FNC 160)

Mnemônico	Função	Operandos					Passos do programa
		S1	S2	S3	S	D	
TCMP FNC 160 (Compara Horário)	Compara dois horários – resultados de <, = e > são dados	K, H, KnX, KnY, KnM, KnS, T, C, D, V, Z			T, C, D	Y, M, S	TCMP, TCMPP: 11 passos
					Nota: 3 endereços consecutivos são usados.		

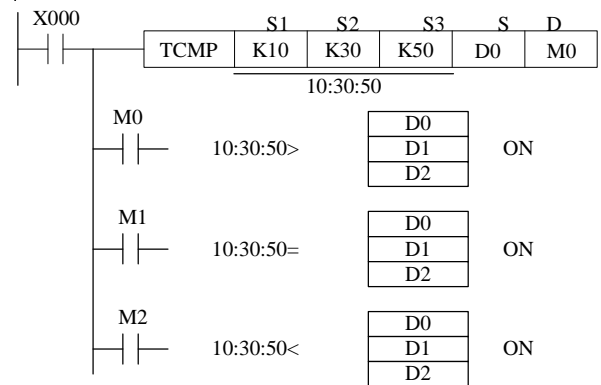
Conteúdo:

S1, S2 e S3 representam horas, minutos e segundos respectivamente. Estes horários são comparados aos valores dos horários nos 3 endereços de dados especificados pelo endereço inicial S. O resultado é indicado nos 3 endereços de bit especificados pelo endereço inicial D.

Os endereços de bit em D indicam o seguinte:

D+0 é ajustado em ON, quando o horário em S é menor que o horário em S1, S2 e S3.

D+1 está ON, quando o horário em S é igual ao horário em S1, S2 e S3.

D+2 está ON, quando o horário em S é maior que o horário em S1, S2 e S3.

Pontos para observar:

a) O status dos endereços de destino é mantido, mesmo que a instrução TCMP seja desativada.

b) A comparação é baseada no valor do horário especificado nos endereços de origem.
- A faixa válida de valores para S1 e S+0 é 0 a 23 (Horas).
- A faixa válida de valores para S2 e S+1 é 0 a 59 (Minutos).
- A faixa válida de valores para S3 e S+2 é 0 a 59 (Segundos).

c) O horário atual do relógio de tempo real pode ser comparado, especificando D8015 (Horas), D8014 (Minutos) e D8013 (Segundos), com os endereços para S1, S2 e S3 respectivamente.

3.14.2 TZCP (FNC 161)

Mnemônico	Função	Operandos				Passos do programa
		S1	S2	S	D	
TZCP FNC 161 (Compara faixa de horário)	Compara um horário a uma faixa de horários especificados – resultados de <, = e > são dados	T, C, D S1 deve ser menor ou igual a S2.			Y, M, S	TZCP, TZCPP: 9 passos
		Nota: 3 endereços consecutivos são usados para todos.				

Conteúdo:

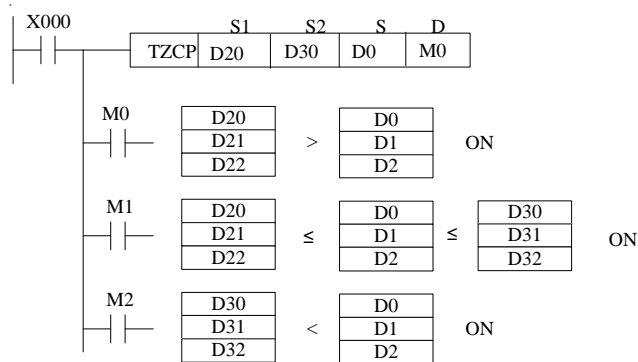
S1, S2 e S representam valores de horas, cada um especificando o endereço inicial de 3 endereços de dados. S é comparado ao período de horários definido por S1 e S2.

O resultado é indicado nos 3 endereços de bits especificados pelo endereço inicial D.

Os endereços de bit em D indicam o seguinte:

D+0 setado quando o horário em S for menor que o horário em S1 e S2.

D+1 setado quando o horário em S estiver entre os horários em S1 e S2.

D+2 setado quando o horário em S for maior que o horário em S1 e S2.

Pontos para observar:

a) O status dos endereços de destino é mantido, mesmo que a instrução TCMP seja desativada.

b) A comparação é baseada no valor do horário especificado nos endereços de origem.

- A faixa válida de valores para S1 e S+0 é 0 a 23 (Horas).
- A faixa válida de valores para S2 e S+1 é 0 a 59 (Minutos).
- A faixa válida de valores para S3 e S+2 é 0 a 59 (Segundos).

3.14.3 TADD (FNC 162)

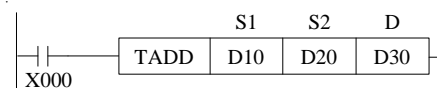
Mnemônico	Função	Operandos			Passos do programa
		S1	S2	D	
TADD FNC 162 (Soma de Horário)	Soma dois valores juntos para dar um novo horário	T, C, D Nota: 3 endereços consecutivos são usados para representar horas, minutos e segundos respectivamente.			TADD, TADDP: 7 passos

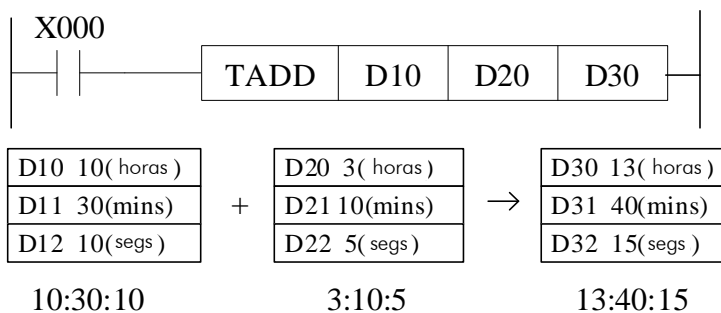
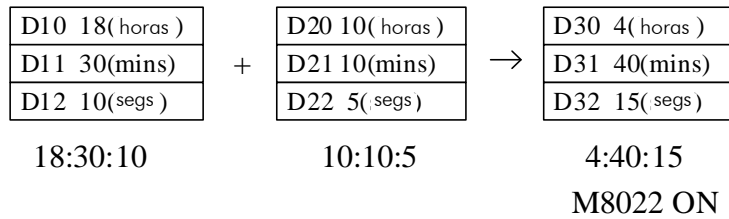
Conteúdo:

Cada S1, S2 e D especificam o endereço inicial de 3 endereços de dados para serem usados um valor de horário.

O valor de horário de S1 é somado ao valor de horário de S2 e o resultado é armazenado em D como um novo valor.

S1 S2 D

X000 TADD D10 D20 D30

Pontos para observar:

a) A soma é executada de acordo com valores padrão de horário. Horas, minutos e segundos são mantidos dentro de limites corretos. Qualquer excesso é processado corretamente.

b) Se a soma dos dois horários resultar em um valor maior que 24 horas, o valor do resultado é o horário remanescente acima de 24 horas.

Quando isso acontece o marcador de carry M8022 é setado.

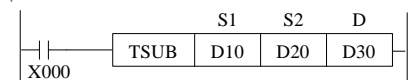
c) Se a soma dos dois horários resultarem num valor igual à zero (0:00:00: 0 horas, 0 minutos, 0 segundos), o marcador M8020 é setado.

d) O mesmo endereço pode ser usado como endereço de fonte (S1 ou S2) e destino. Neste caso, a soma é executada continuamente; o valor de destino mudando a cada varredura de programa. Para evitar que isso aconteça, use uma habilitação por pulso ou um intertravamento no programa.

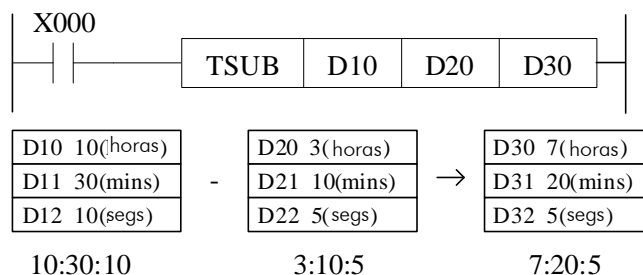
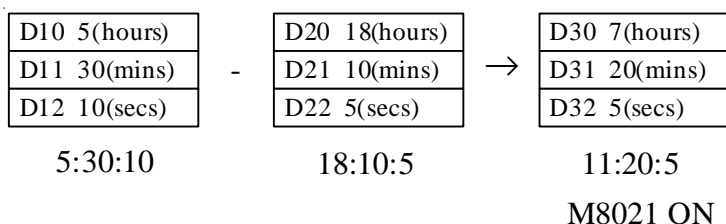
3.14.4 TSUB (FNC 163)

Mnemônico	Função	Operandos			Passos do programa
		S1	S2	D	
TSUB FNC 163 (Subtração de Horário)	Subtrai um valor de horário de outro valor de horário e fornece um novo horário	T, C, D Nota: 3 endereços consecutivos são utilizados.			TSUB, TSUBP: 7 passos

Conteúdo:

Cada S1, S2 e D especifica o endereço inicial de 3 endereços de dados para usarem um valor de horário.

O valor do horário S1 é subtraído do valor de horário em S2, e o resultado é armazenado em D como um novo valor de horário.

Pontos para observar:

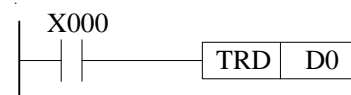
a) A subtração é efetuada de acordo com valores de horário padrão. Horas, minutos e segundos são mantidos dentro de limites corretos. Qualquer excesso é corretamente processado.

a) Se a subtração de dois horários resultar um valor menor que 00:00:00, o resultado será a diferença entre 00:00:00 e o resultado da subtração entre S2 e S1. O marcador especial M8021 será ativado.

b) Se a subtração de dois horários resultar o valor zero, o marcador especial M8020 será ativado.

c) Os mesmos registradores podem ser utilizados como fonte e destino na mesma função TSUB, porém dessa forma terão seus valores alterados a cada ciclo de scan. Para evitar este problema podem ser elaboradas lógicas de intertravamento, ou utilizar a função TSUBP, que é executada por borda de subida.

3.14.5 TRD (FNC 166)

Mnemônico	Funcão	Operandos	Passos do programa
		D	
TRD FNC 166 (Leitura de Tempo)	Transfere o valor atual do RTC para um grupo de registradores	T, C, D Nota: 7 endereços consecutivos são utilizados	TRD, TRDP: 5 passos

Operação:

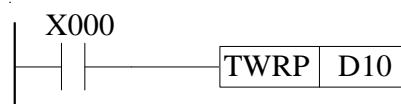
Quando a função TRD é habilitada, os valores atuais de data e hora do RTC são lidos e transferidos para os 7 registradores definidos a partir de D.

Descrição dos registradores utilizados:

Endereço	Significado	Valor		Endereço	Significado
D8018	Ano	2000~2099	→	D0	Ano
D8017	Mês	1~12	→	D1	Mês
D8016	Dia	1~31	→	D2	Dia
D8015	Hora	0~23	→	D3	Horas
D8014	Minuto	0~59	→	D4	Minuto
D8013	Segundos	0~59	→	D5	Segundos
D8019	Dia da semana	0(dom)~6(sáb)	→	D6	Dia da semana

3.14.6 TWR (FNC 167)

Mnemônico	Função	Operandos	Passos de programa
		S	
TWR FNC 167 (Ajuste do RTC)	Define o valor do RTC	T, C, D Nota: 7 endereços consecutivos são utilizados	TWR, TWRP: 5 passos

Operação:

Quanto a função TWR é ativada, os 7 registradores consecutivos especificados em S serão transferidos para ajuste do RTC, ajustando seu novo valor.

Descrição dos registradores utilizados:

Endereço	Significado	Valor		Endereço	Significado
D10	Ano	0~99	→	D8018	Ano
D11	Mês	1~12	→	D8017	Mês
D12	Dia	1~31	→	D8016	Dia
D13	Hora	0~23	→	D8015	Hora
D14	Minuto	0~59	→	D8014	Minuto
D15	Segundos	0~59	→	D8013	Segundos
D16	Dia da semana	0(dom)~6(sáb)	→	D8019	Dia da semana

Pontos para observar:

Esta função não utiliza o marcador M8015 (ver marcadores de sistema) para definir um novo valor para o RTC, facilitando seu ajuste.

3.15 CÓDIGOS CINZAS - FNC 170 A FNC 179

Conteúdo:

GRY - Conversão Decimal para Código "Gray" FNC 170
GBIN - Conversa Código "Gray" para Decimal FNC 171
Não disponível FNC 172 a 177

Lista de símbolos:

D – Endereço de destino.

S – Endereço de origem.

m, n - Número de endereços ativos, bits ou uma constante operacional.

Sufixos numéricos adicionais serão anexados se houver mais de um operando com a mesma função, ex: D1, S3 ou para endereços de lista/tabelados D3+0, S+9 etc.

MSB – Bit mais significante, por vezes usado para indicar o sinal matemático de um número, ex: positivo = 0, e negativo = 1.

LSB – Bit menos significante.

Modificações de instrução:

□□□- Instrução de 16 bits, onde □□□ identifica a instrução mnemônica.
□□□P- Instrução de 16 bits habilitada por pulso (simples).
D□□□- Instrução de 32 bits.
D□□□P - Instrução de 32 bits habilitada por pulso (simples).

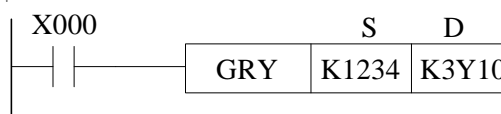
★ – Uma instrução repetitiva que mudará o valor de destino a cada varredura menos quando for habilitada por um pulso.

☆ – Um operando que não pode ser indexado, ex: A adição de V ou Z ou é inválida ou não terá efeito para o valor do operando.

3.15.1 GRY (FNC 170)

Mnemônico	Função	Operandos		Passos do programa
		S	D	
GRY FNC 170 (Código Gray)	Converte um valor decimal em código "gray".	K, H, KnX, KnY, KnM, KnS, T, C, D, V, Z	KnY, KnM, KnS, T, C, D, V, Z	GRY,GRYP: 5 passos DGRY,DGRYP 9 passos

Operação:

O valor da integral binário em S é convertido para o código "Gray" equivalente e armazenado em D.

Pontos para observar:

A utilização dos números com código "gray" permite que valores numéricos sejam enviados rapidamente para saídas sem a necessidade de uma habilitação. Por exemplo, se os dados de origem forem continuamente incrementados, os novos dados de saída podem ser atualizados a cada varredura do programa.

3.15.1 GBIN (FNC 171)

Mnemônico	Função	Operandos		Passos do programa
		S	D	
GBIN FNC 171 (Código "Gray")	Converte um valor no cógigo "gray" para decimal	K, H, KnX, KnY, KnM, KnS, T, C, D, V, Z	KnY, KnM, KnS, T, C, D, V, Z	GBIN,GBINP: 5 passos DGBIN,DGBINP: 9 passos

Operação:

O valor em código "gray" em S é convertido um valor binário normal equivalente e armazenado em D.

Pontos para observar:

Esta instrução pode ser usada para ler o valor de um encoder de código "gray".

Se a origem for ajustada para as saídas X0 a X17, é possível apressar o tempo de leitura ajustando o filtro de atualização com FNC 51 REFF.

3.16 CÓDIGOS DE COMUNICAÇÃO - FNC 190 À FNC 199

Conteúdo:

DTLK -	Link de Dados	FNC 190
RMIO -	IO Remoto	FNC 191
TEXT-	OP07/08 TEXT	FNC 192
□□□-	Não Disponível	FNC 193 to 199

Lista de símbolos:

D – Endereço de destino.

S – Endereço de origem.

m, n - Número de endereços ativos, bits ou uma constante operacional.

Sufixos numéricos adicionais serão anexados se houver mais de um operando com a mesma função, ex: D1, S3 ou para endereços de lista/tabelados D3+0, S+9 etc.

MSB – Bit mais significante, por vezes usado para indicar o sinal matemático de um número, ex: positivo = 0, e negativo = 1.

LSB – Bit menos significante.

Modificações de instrução:

□□□- Instrução de 16 bits, onde □□□ identifica a instrução mnemônica.

□□□P - Instrução de 16 bits habilitada por pulso (simples).

D□□□ - Instrução de 32 bits.

D□□□P - Instrução de 32 bits habilitada por pulso (simples).

★ – Uma instrução repetitiva que mudará o valor de destino a cada varredura menos quando for habilitada por um pulso.

☆ – Um operando que não pode ser indexado, ex: A adição de V ou Z ou é inválida ou não terá efeito para o valor do operando.

3.16.1 DTLK (FNC 190)

Mnemônico	Função	Operandos	Passos do programa
		K	
DTLK FNC 190 (Link de dados)	Monta uma pequena rede que permite que um CLP controle outros 15 CLPs.	K,H:0,1 0: Porta de comunicação RS485 integrada; 1: Portas na placa de expansão RS485 ou RS232	3 passos

Operação:

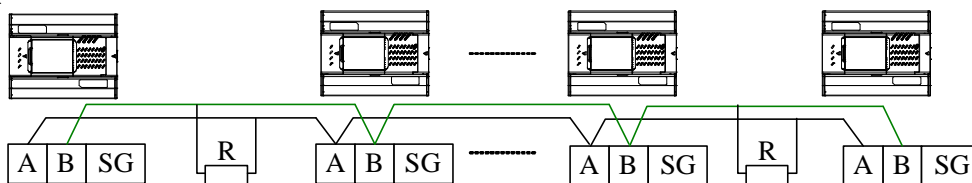
Esta instrução F190 DTLK, usada pelo CLP, pode monta uma pequena rede que permite que um CLP controle outros 15 CLPs.

Apesar de duas portas de comunicação estarem prontos para DTLK, somente a que for habilitada primeiro é que executará a comunicação.

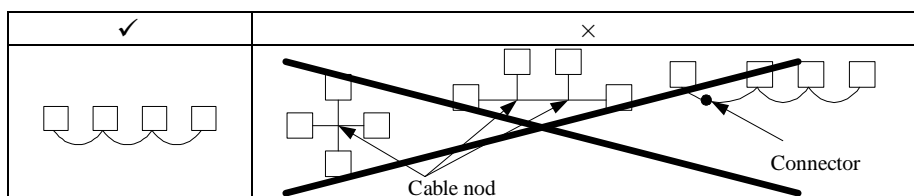
O formato do telegrama e o baud rate são ajustados através de D8120 ou D8320, cada um para uma porta diferente.

Ambas as portas RS485/ RS232 das placas de expansão (todos os tipos estão disponíveis para expansão)e a porta integrada RS485 (existente somente no modelo H) podem ser utilizadas para comunicação DATA LINK. Porém, elas não podem ser habilitadas simultaneamente.

Item	Especificação
Padrão de comunicação	EIA RS-485
Baud Rate	9600bps～307200bps
Número de escravos	Máx 15 escravos
Endereços para comunicação	D0～D157，M2000～M3023
Comprimento dos dados para cada escravo	Máx 64 bits+8 palavras
Cabo de comunicação	Par trançado com malha, duas vias, comprimento total: 500m (76800bit/s), 1km(38400bit/s).

Instalação elétrica:

☑ Nota 1: O borne a ser usado deve ser específico para comunicação, ou a mesma poderá ser interrompida por causa de ruído.

☑ Nota 2: Ramificação do cabo de conexão não deve exceder 3.

☑ Nota 3: R representa o resistor de terminação (120Ω, 1/4W).

Endereços para comunicação:

1) Marcadores especiais

Marcadores especiais	Característica	Função	Descrição	Resposta de
M8400	Somente leitura	Erro no Mestre	O bit estará setado enquanto o mestre estiver com erro	L
M8401	Somente leitura	Erro no Escravo 1	O bit estará setado enquanto o escravo 1 estiver com erro.	M/L
M8402	Somente leitura	Erro no Escravo 2	O bit estará setado enquanto o escravo 2 estiver com erro.	M/L
…	…	…	…	..
M8414	Somente leitura	Erro no Escravo 14	O bit estará setado enquanto o escravo 14 estiver com erro.	M/L
M8415	Somente leitura	Erro no Escravo 15	O bit estará setado enquanto o escravo 15 estiver com erro.	M/L
M8416	Somente leitura	Estado	O bit estará setado enquanto DTLK estiver habilitada.	M/L
M8417	Somente leitura	Modo DATA LINK	O bit estará setado enquanto a placa de expansão estiver no DATA LINK	M/L
M8418	Somente leitura	Modo DATA LINK	O bit estará setado enquanto a porta RS485 estiver no DATA LINK	M/L

2) Registro de dados

Marcadores especiais	Característica	Função	Descrição	Resposta de
D8173	Somente leitura	Número do nó	Armazena o próprio número do PLC na rede	M/L
D8174	Somente leitura	Número de escravos	Armazena o número de escravos	M/L
D8175	Somente leitura	Taxa de atualização	Armazena da taxa de atualização do dados na DATA LINK	M/L
D8176	Escrita	Define o endereço do Escravo	Define o número do próprio PLC	M/L
D8177	Escrita	Define o número de Escravos	Define o número de escravos na rede	M
D8178	Escrita	Define DATA LINK	Define taxa de atualização	M
D8179	Leitura/ Escrita	Número de novas tentativas	Define o número de tentativas no caso de falha de transmissão	M
D8180	Leitura/ Escrita	Define o tempo de “time-out”	Define o tempo máximo para que uma transmissão seja realizada	M
D8401	Somente leitura	Valor atual do ciclo de varredura da rede de comunicação	Armazena o valor do ciclo de varredura da comunicação DATA LINK	M/L
D8402	Somente leitura	Máx ciclo de varredura da comunicação	Armazena o valor máx medido do ciclo de varredura da rede	M/L
D8403	Somente leitura	Número de erros no mestre	Número de erros ocorridos no mestre	L
D8404	Somente leitura	Número de erros no escravo 1	Número de erros no escravo 1	M/L
D8405	Somente leitura	Número de erros no escravo 2	Número de erros no escravo 2	M/L
…	…	…	…	..
D8411	Somente leitura	Número de erros no escravo 8	Número de erros no escravo 8	M/L
…	…	…	…	..
D8417	Somente leitura	Número de erros no escravo 14	Número de erros no escravo 14	M/L
D8418	Somente leitura	Número de erros no escravo 15	Número de erros no escravo 15	M/L
D8419	Somente leitura	Código do erro do mestre	Código de erro do mestre	L
D8420	Somente leitura	Código do erro do escravo 1	Código de erro do escravo 1	M/L
D8421	Somente leitura	Código do erro do escravo 2	Código de erro do escravo 2	M/L
…	…	…	…	..
D8427	Somente leitura	Código do erro do escravo 8	Código de erro do escravo 8	M/L
…	…	…	…	..
D8433	Somente leitura	Código do erro do escravo 14	Código de erro do escravo 14	M/L
D8434	Somente leitura	Código do erro do escravo 15	Código de erro do escravo 15	M/L

Ajuste:

Quando o programa estiver em operação, ou TP-03 estiver energizado, todos os ajustes da DATA LINK irão tomar efeito.

1) Ajustando o endereço do escravo (D8176)

Ajuste 0~15 para o registro de dados especial D8176, 0 é para o mestre, e 1~ é para o escravo.

2) Ajustando o número dos escravos (D8177)

Ajuste 0~15 para o registro de dados especial D8177(padrão: 7). Não é necessário para escravos.

O número dos escravos deve ser ajustado de acordo com as diferentes condições para aumentar a velocidade de atualização da rede.

3) Ajustando a taxa de atualização (D8178)

Ajuste 0~2 para o registro de dados especial D8178 (padrão: 0). Não é necessário para escravos.

D8178		0	1	2
Modo DATA LINK		Modo 0	Modo 1	Modo 2
Taxa de atualização	Endereço de bit (M)	0 pontos	32 pontos	64 pontos
	Endereço de word (D)	4 pontos	4 pontos	8 pontos

Os endereços a serem atualizados sob modo diferente:

Endereço	Modo 0		Modo 1		Modo 2	
	(M)	(D)	(M)	(D)	(M)	(D)
No 0	—	D0~D3	M2000~M2031	D0~D3	M2000~M2063	D0~D7
No 1	—	D10~D13	M2064~M2095	D10~D13	M2064~M2127	D10~D17
No 2	—	D20~D23	M2128~M2159	D20~D23	M2128~M2191	D20~D27
No 3	—	D30~D33	M2192~M2223	D30~D33	M2192~M2255	D30~D37
No 4	—	D40~D43	M2256~M2287	D40~D43	M2256~M2319	D40~D47
No 5	—	D50~D53	M2320~M2351	D50~D53	M2320~M2383	D50~D57
No 6	—	D60~D63	M2384~M2415	D60~D63	M2384~M2447	D60~D67
No 7	—	D70~D73	M2448~M2479	D70~D73	M2448~M2511	D70~D77
No 8	—	D80~D83	M2512~M2543	D80~D83	M2512~M2575	D80~D87
No 9	—	D90~D93	M2576~M2607	D90~D93	M2576~M2639	D90~D97
No A	—	D100~D103	M2640~M2671	D100~D103	M2640~M2703	D100~D107
No B	—	D110~D113	M2704~M2735	D110~D113	M2704~M2767	D110~D117
No C	—	D120~D123	M2768~M2799	D120~D123	M2768~M2831	D120~D127
No D	—	D130~D133	M2832~M2863	D130~D133	M2832~M2895	D130~D137
No E	—	D140~D143	M2896~M2927	D140~D143	M2896~M2959	D140~D147
No F	—	D150~D153	M2960~M2991	D150~D153	M2960~M3023	D150~D157

4) Ajustando número de novas tentativas (D8179)

Ajuste 0~10 para o registro de dados especial D8179 (padrão: 3). Não é necessário para escravos. Se o mestre tentar se comunicar com o escravo novamente acima do número de vezes ajustadas, o escravo entrará em erro de comunicação.

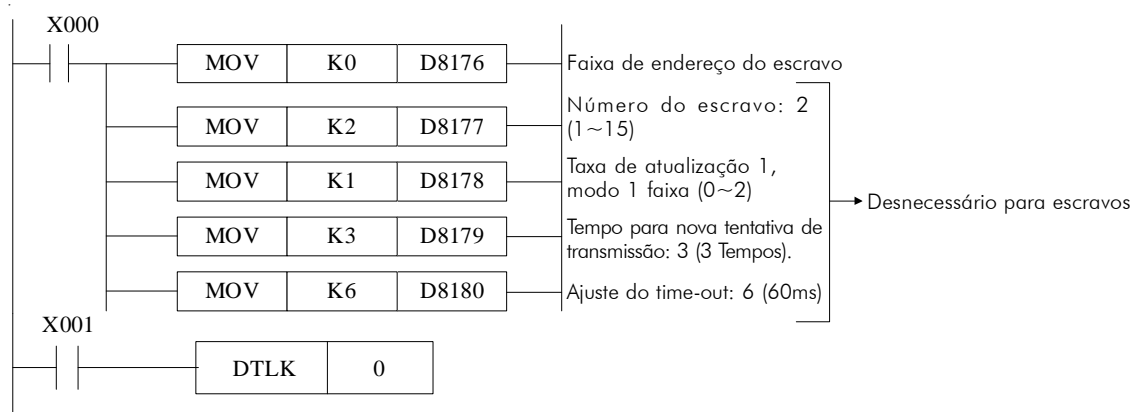
5) Ajuste de time-out (D8180)

Ajuste 5~255 para o registro de dados especial D8180 (padrão: 5), o produto de tal valor vezes 10 é o tempo máximo que CLP tem para realizar a transmissão de dados na rede (ms).

6) Valor atual do ciclo de varredura da rede (D8401)

O produto de tal valor vezes 10 é o ciclo atual de varredura da comunicação (ms).

7) Ciclo de varredura máximo (D8402)

Exemplo de programa para ajustar o tal dispositivo:

Código de erro:

Quando há um erro, os marcadores especiais M8400~M8415 irão indicar a condição de erro e o código do erro será armazenado em registros de dados especiais (D8419~D8434).

Código do erro	Erro	Endereço do erro	Verificação de endereço	Descrição	Ponto de verificação
01H	Erro de comunicação (time-out)	L	M	Não há resposta quando o mestre manda um telegrama para o escravo.	Fiação, fonte de alimentação e estado de run/ stop
02H	Erro de comunicação (número do nó inválido)	L	M	O endereço não foi ajustado de acordo com certas relações entre mestre e escravo	Fiação
03H	Erro na contagem da comunicação	L	M	Os dados no contador de comunicação não estão de acordo com certas relações entre mestre e escravo	Fiação
04H	Erro no formato do telegrama de comunicação	L	M, L	Erro no formato do telegrama de comunicação	Fiação e ajuste de DTLK
11H	Erro de comunicação (Tempo de comunicação excedido)	M	L	Depois das respostas do escravo para o mestre, o mestre não envia outro pedido para os escravos	Fiação, fonte de alimentação e estado de run/ stop
14H	Erro no telegrama de comunicação	M	L	Erro na carcaça da comunicação do mestre	Fiação e ajuste de DTLK
21H	Sem escravo	L	L *1	Endereço da rede incorreto	Ajuste de endereço
22H	Erro no endereçamento	L	L *1	Endereço do escravo não está de acordo com certas relações entre mestre e escravo	Fiação
23H	Erro na contagem da comunicação	L	L *1	Os dados no contador da comunicação não estão de acordo com certas relações entre mestre e escravo	Fiação
31H	Erro no parâmetro da comunicação de recebimento	L	L *2	Mestre envia pedido antes do escravo aceitar o parâmetro ajustado	Fiação, fonte de alimentação e estado de run/ stop
32H	Outros erros	L	L *1	Erro na instrução de comunicação	Definição dos parâmetros da rede

M: mestre L: escravo
1: outro escravo 2: Escravo individual

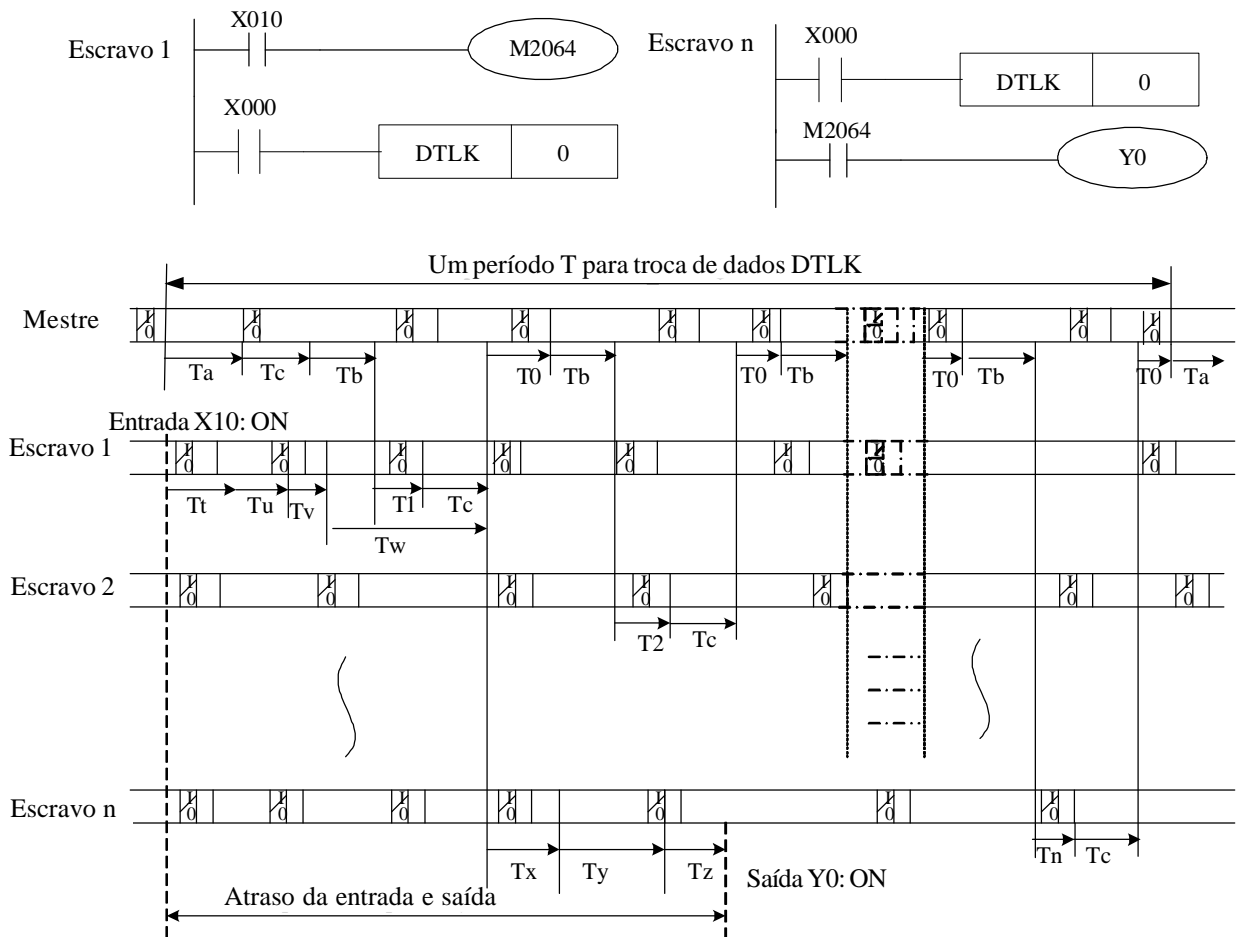
Temporização da seqüência de comunicação e tempo necessário para transmissão.

A comunicação entre mestre-estação e escravo-estação não é síncrona ao ciclo de varredura do mestre-estação.

O mestre-estação irá executar a troca de dados e atualizar o marcador de comunicação no ciclo de varredura depois de completar a comunicação.

Diagrama da temporização da seqüência de comunicação e do atraso de comunicação.

Na rede DATA LINK, haverá um atraso para recebimento dos dados. Favor referir-se à seguinte figura para temporização da seqüência de comunicação:

Por exemplo: o M2064 para escravo 1 é controlado por X010. O estado de M2064 será enviado para um outro nó da rede quando a instrução DTLK for habilitada.

Tempo necessário para completar a transmissão

No modo de comunicação DATA LINK, o tempo que T necessário para completar a comunicação mestre-estação com todos os escravos-estações pode ser descrito da seguinte maneira (não utilizará o CICLO DE VARREDURA do mestre-estação):

T=Ta+Tc+[Tb+Tn+Tc+T0]*n1{*+[Tb+Tn+D8180*10]*n2*}:

T_a : o tempo de transmissão da instrução de envio do mestre para a configuração de rede do escravo.

T_b :o tempo de transmissão da instrução de envio do mestre para a troca de dados do escravo.

T_c : o tempo de transmissão para a troca de dados da rede (difere do modo DTLK diferente).

T_0 :o tempo do estado de comunicação de detecção do mestre (0~1 CICLO DE VARREDURA)

T_n: o tempo do estado de comunicação de detecção do escravo (0~1 CICLO DE VARREDURA)

(n1+n2): número do escravo DTLK ajustado no mestre (D8177=1~15), n1: número atual do escravo, n2: número do escravo que não é reconhecido pelo mestre (0~15).

D8180 é o valor de time-out.

Tempo de atraso:

Tu: o tempo necessário para o CLP detectar o status de entrada (máx. 1 CICLO DE VARREDURA);

Tv: o tempo entre o estado da entrada ser recebido pelo CLP e o programa iniciar a varredura;

Tw: o tempo para o envio do resultado de operação (máx. ciclo de varredura da rede);

Tx: tempo entre dados recebidos e dados escritos para os registros (máx. 1 ciclo de varredura);

Ty: o tempo entre a operação do programa e a saída (1 ciclo de varredura);

Tz: atraso da porta de comunicação

O tempo de transmissão conforme o baud rate selecionado:

Baud Rate (bps)	Ta（ms）	Tb（ms）	Tc（ms）		
			Modo DTLK 0	Modo DTLK 1	Modo DTLK 2
9600	21,8	12,6	31,0	40,1	67,6
19200	10,9	6,3	15,5	20,1	33,8
38400	5,5	3,2	7,8	10,0	16,9
57600	3,7	2,1	5,2	6,7	11,3
76800	2,8	1,6	3,9	5,0	8,5
128000	1,7	1,0	2,4	3,0	5,1
153600	1,4	0,8	2,0	2,5	4,3
307200	0,7	0,4	1,0	1,3	2,2

3.16.2 RMIO (FNC 191)

Mnemônico	Função	Operandos K	Passos do programa
RMIO FNC 191 (IO Remoto)	Montar uma pequena rede que permite que um CLP controle outros 4 CLPs.	K, H:0,1 0: Porta RS485 integrada à CPU; 1: Porta RS485 ou RS232 na placa de expansão.	3 passos

Operação:

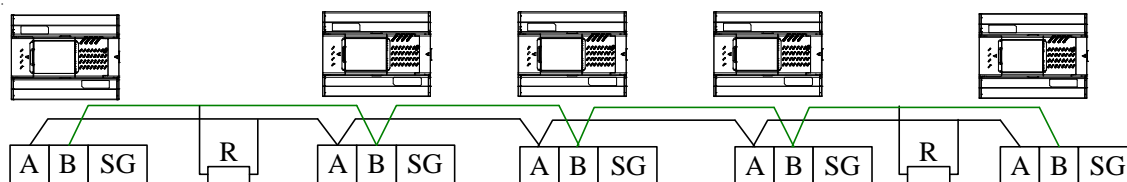
Esta função F191 RMIO utilizada pela CLP permite montar uma pequena rede que conecta o CLP à controlar outros 4 CLPs.

Apesar de haverem duas portas de comunicação disponíveis para a rede RMIO, somente aquela que for habilitada primeiro será utilizada na comunicação.

O formato do telegrama de comunicação e a taxa de transmissão são parâmetros ajustados pelos endereços D8120 ou D8320.

☑ Nota 1: Quando um CLP é ajustado como escravo na rede RMIO, ele passa a ser utilizado unicamente como expansão de I/O do CLP mestre, todo o restante do programa não será executado.

☑ Nota 2: Uma vez habilitada a instrução RMIO em CLP configurado como escravo, somente um STOP no CLP poderá desabilitá-la.

Na rede I/O Remoto, o CLP mestre pode controlar outros 4 CLPs.

Item	Descrição	
Padrão de comunicação	EIA RS485	
Taxa de Transmissão	9600bps～307200bps	
Número de escravos	Máx 4 escravos	
Endereços relacionados	Escravo 1	Entrada: 36 pontos (M4200～M4235); Saída: 24 pontos (M4600～M4623)
	Escravo 2	Entrada: 36 pontos (M4240～M4275); Saída: 24 pontos (M4624～M4647)
	Escravo 3	Entrada: 36 pontos (M4280～M4315); Saída: 24 pontos (M4648～M4671)
	Escravo 4	Entrada: 36 pontos (M4320～M4355); Saída: 24 pontos (M4672～M4695)
Cabo	Par trançado com malha, 2 vias Comprimento total: 500m (76800bit/s), 1km(38400bit/s)	

As placas de expansão de ambas as portas, RS485/ RS232 (todos os tipos estão disponíveis para expansão) a porta RS485 (porta integrada no módulo base da CPU), estão disponíveis para a comunicação I/O remoto. Porém, elas não podem ser habilitadas simultaneamente.

Nota: Somente unidade básica pode ser ajustada como escravo em modo RMIO.

Endereços relacionados:

1) Marcadores especiais

Marcadores especiais	Característica	Função	Descrição	Responder de
M8335	Somente leitura	Status da Comunicação	Setado enquanto comunicação RMIO está habilitada	M/L
M8336	Somente leitura	Erro no Mestre	Setado quando houver um erro no Mestre	L
M8337	Somente leitura	Erro no Escravo 1	Setado quando houver um erro no Escravo 1	M/L
M8338	Somente leitura	Erro no Escravo 2	Setado quando houver um erro no Escravo 2	M/L
M8339	Somente leitura	Erro no Escravo 3	Setado quando houver um erro no Escravo 3	M/L
M8340	Somente leitura	Erro no Escravo 4	Setado quando houver um erro no Escravo 4	M/L
M8341	Somente leitura	Modo RMIO	Placa de expansão em modo RMIO	M/L
M8342	Somente leitura	Modo RMIO	Porta RS485 em modo RMIO	M/L

2) Registro de dados D

Marcadores especiais	Característica	Função	Descrição	Responde
D8373	Somente leitura	Número de endereço	Armazena o próprio endereço do PLC	M/L
D8374	Somente leitura	Número de escravos	Armazena o número de escravos	M/L
D8376	Escrita	Define o número do nó	Ajusta o endereço do próprio PLC	M/L
D8377	Escrita	Define o número de escravos	Ajusta o número de escravos na rede	M
D8379	Leitura/Escrita	Número de novas tentativas	Ajusta o número de tentativas em caso de falha de comunicação	M
D8380	Leitura/Escrita	Ajuste de time-out	Ajusta o tempo máximo de transmissão dos dados da rede	M/L
D8331	Somente leitura	Ciclo atual de varredura da rede	Armazena o valor atual do ciclo de varredura da rede	M
D8332	Somente leitura	Ciclo de varredura máximo	Armazena o valor máximo medido do ciclo de varredura da rede	M
D8333	Somente leitura	Número de erros no mestre	Número de erros no mestre	L
D8334	Somente leitura	Número de erros no escravo 1	Número de erros no escravo 1	M/L
D8335	Somente leitura	Número de erros no escravo 2	Número de erros no escravo 2	M/L
D8336	Somente leitura	Número de erros no escravo 3	Número de erros no escravo 3	M/L
D8337	Somente leitura	Número de erros no escravo 4	Número de erros no escravo 4	M/L
D8338	Somente leitura	Código do erro do mestre	Código do erro do mestre	L
D8339	Somente leitura	Código do erro do escravo 1	Código do erro do escravo 1	M/L
D8340	Somente leitura	Código do erro do escravo 2	Código do erro do escravo 2	M/L
D8341	Somente leitura	Código do erro do escravo 3	Código do erro do escravo 3	M/L
D8342	Somente leitura	Código de erro do escravo 4	Código do erro do escravo 4	M/L

Ajuste:

Quando o programa estiver em operação, ou o CLP estiver energizado, todos os ajustes para I / O Remoto terão efeito.

1) Ajustando o endereço do escravo (D8376)

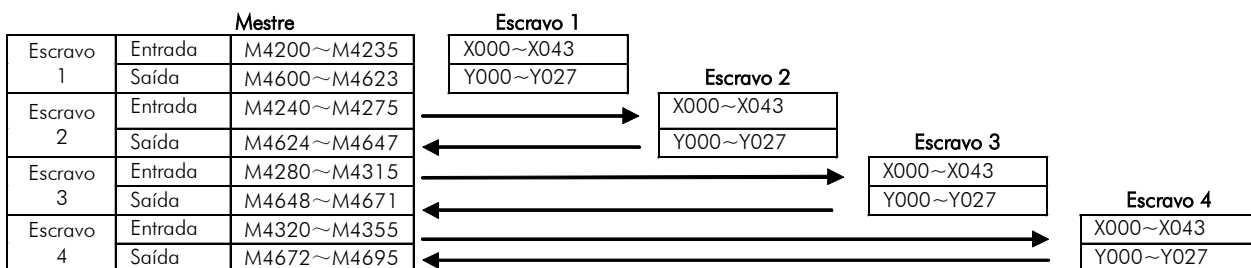
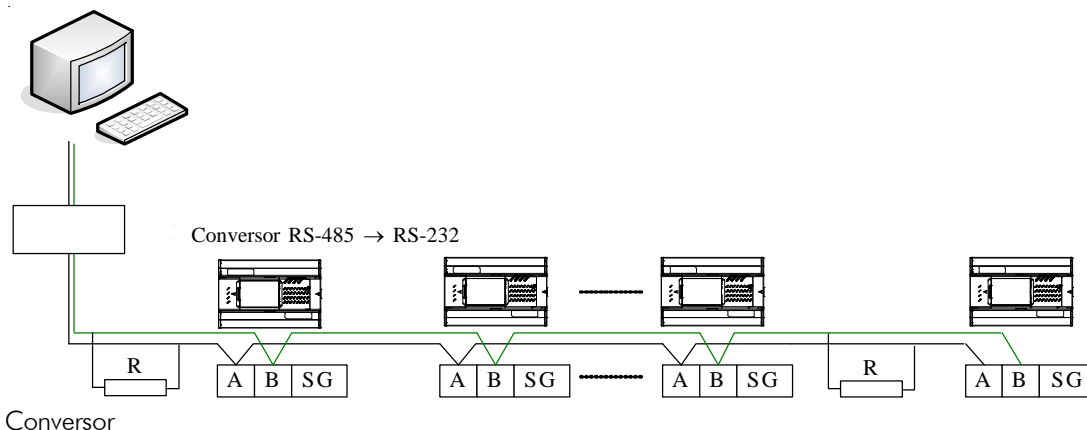
Ajuste 0～4 para o registro de dados especial D8376, 0 é para o mestre, e 1～4 é para o escravo.

2) Ajustando o número de escravos (D8377)

Ajuste 1～4 para o registro de dados especial D8377 (padrão: 4). Não é necessário para escravos.

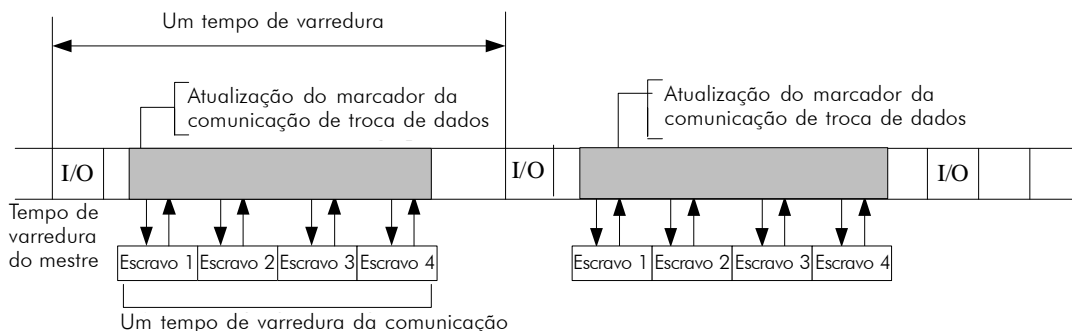
O número de escravos deve ser ajustado de tal forma a tentar minimizar o tempo de atualização da rede.

Os endereços relacionados à rede I/O Remoto:

Na rede I/O Remoto, os endereços relacionados para o mestre:

Fiação:

☑ Nota 1: O borne a ser utilizado deve ter características tais de modo a evitar que a comunicação da rede venha a ser interrompida influência de ruídos externos.

☑ Nota 2: Ramificação do cabo de comunicação não deve exceder 3.

☑ Nota 3: R representa resistor de terminação (120Ω,1/4W).

Seqüência de comunicação e o tempo necessário para transmissão

O Tempo Necessário para Transmissão

Na comunicação da estação mestre para estação escrava, a troca de dados na rede I/O remoto e a atualização do marcador de comunicação são síncronos com o ciclo de varredura da estação mestre. O processo de comunicação irá aumentar o ciclo de varredura da estação mestre.

Quando há um erro na comunicação entre mestre e escravo, a comunicação I/O remoto e operação CLP irão parar e entrar em condição anormal.

Quando ocorre um erro na comunicação entre a estação mestre e a estação escrava, a comunicação I/O remota e a execução do programa do CLP serão paradas e irão entrar em modo de erro.

Além disso, todos os marcadores de comunicação da estação mestre e estação escrava serão zerados.

Possíveis causas de erro são as seguintes:

① Erro no CRC

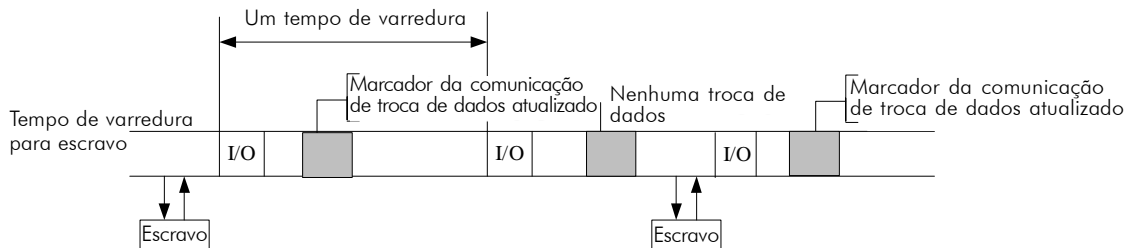
② Escravo em modo de STOP ou modo de ERRO

③ Escravo não conectado ou fio de conexão partido

Quando a estação mestre estiver no modo STOP ou modo ERRO, ela não se comunicará com nenhuma estação escravo. Os ajustes do telegrama de comunicação entre mestre e escravo não são iguais.

Seqüência de comunicação para escravo

A comunicação do escravo para o mestre é assíncrona ao ciclo de varredura do escravo.

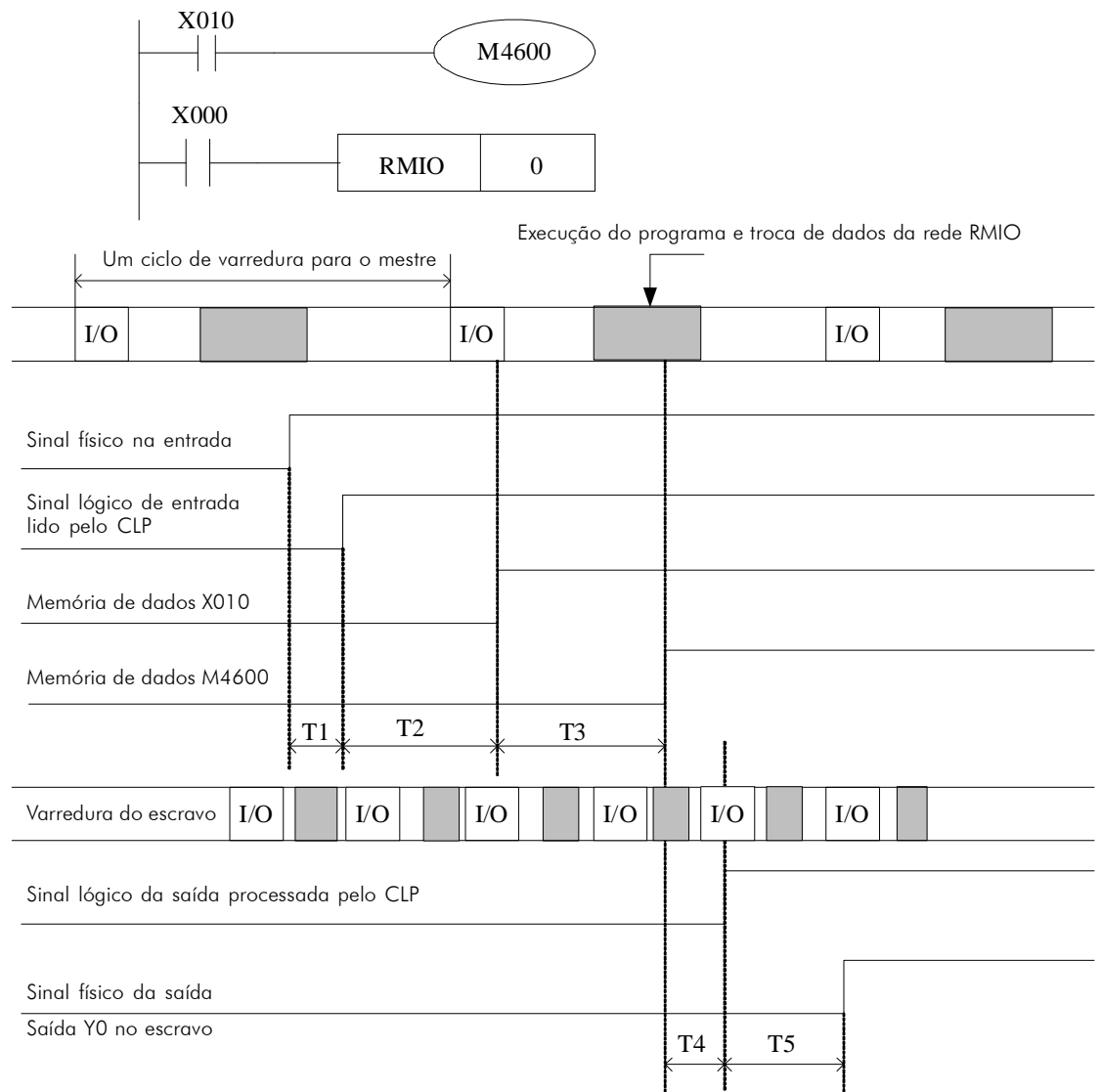
Após encerrar a comunicação entre mestre e escravo, os dados I/O Remotos e marcador de comunicação serão atualizados, o que irá durar aproximadamente 0,2ms.

O tempo necessário para a transmissão

Na rede I/O remoto, o tempo T (período de comunicação a ser acrescido ao tempo do ciclo de varredura da estação mestre) necessário para a estação mestre completar a comunicação com todas as estações escravas é o seguinte:

Taxa de transmissão (bps)	Tempo de comunicação para cada escravo. Tn (ms)	Time out t (ms)	Tempo de comunicação para o mestre, T(ms)	Tempo de comunicação normal para mestre e 4 escravos (ms)
9600	42			168
19200	21			84
38400	11			44
57600	7			28
76800	6	D8380*10	Tn*n1＋t*n2 (n1: número normal de escravos; n2: número de escravos para time out).	24
128000	4			16
153600	3			12
307200	2			8

Se houver um erro de comunicação no escravo, o tempo de comunicação será aumentado repetidamente. (Tn será adicionado ao tempo para cada erro) Taxa de transmissão.

Tempo de atraso:

Quando o I/O remoto estiver recebendo dados, haverá um atraso conforme figura a seguir:

T1: Atraso da entrada (tempo para leitura do estado da entrada)
T2: tempo para o mestre escrever os dados no registrador de dados
T3: Execução do programa do usuário e atualização das saídas
T4: tempo para o escravo receber o dado transmitido
T5: Atraso da atualização da saída (tempo de resposta para OFF a ON)

Código de erro:

Quando há um erro, os relés especiais M8336~M8340 indicarão a condição de erro e o código de erro será armazenado em registros de dados especiais D8338~D8342.

Código do erro	Erro	Endereço do erro	Verificação de endereço	Descrição	Ponto de verificação
01H	Erro de comunicação (time-out)	L	M	Não há resposta quando o mestre manda um telegrama para o escravo	Fiação, fonte de alimentação e estado de run/ stop
02H	Erro de comunicação (endereçamento)	L	M	O endereço não foi ajustado corretamente	Fiação
03H	Erro na contagem da comunicação	L	M	Os dados no contador de comunicação não estão de acordo com certas relações entre mestre e escravo	Fiação
04H	Erro no formato do telegrama	L	M, L	Erro no formato do telegrama de comunicação com o escravo	Fiação e ajuste de RMIO
11H	Erro de comunicação (falha na transmissão)	M	L	Depois das respostas do escravo para o mestre, o mestre não envia outro pedido aos escravos	Fiação, fonte de alimentação e estado de run/ stop
14H	Erro no formato do telegrama	M	L	Erro no formato do telegrama de comunicação com o mestre	Fiação e ajuste de RMIO
21H	Sem escravo	L	L *1	Endereço da rede incorreto	Ajuste de endereço
22H	Erro de endereçamento	L	L *1	Endereço do escravo ajusta incorretamente	Fiação
23H	Erro na contagem da comunicação	L	L *1	Os dados no contador da comunicação não estão de acordo com certas relações entre mestre e escravo	Fiação
24H	Erro no formato do telegrama	L	L *1	Erro no formato do telegrama com o escravo	Fiação e ajuste de RMIO

M: mestre L: escravo

*1: outro escravo

3.16.3 TEXT (FNC 192)

Mnemônico	Função	Operandos			Passos do programa
		D	S	n	
TEXT FNC 192	Texto do display (incluindo dados de registro) no LCD OP08	D	D	K, H: 1,2	7 passos

Operação:

Esta instrução deve ser utilizada com a interface homem-máquina (IHM) OP08. Depois de habilitar F192, o valor 13 será escrito no registro de dados D8284, depois da OP08 armazenar o dado '13' em D8284, o valor 13 também será escrito em D8285 pela OP08.

Conforme F192 é habilitado, um determinado arquivo de texto será armazenado em D8280 e D8281 (D8280 é para o arquivo ser apresentado na primeira linha da OP08, D8281 é para a segunda linha) e o valor da variável ser apresentada será armazenado em D8295 e D8296.

O valor em D8295 será apresentado na posição '#' da primeira linha, enquanto o valor em D8296 será mostrado na segunda.

Onde houver um '?' no display de LCD, dados poderão ser inseridos. Os dados inseridos na primeira linha serão armazenados no registro D (Número = valor em D8295 + 1). Em relação à segunda linha, os dados inseridos na posição '?' serão armazenados no registro D (Número =valor em D8296 + 1).

'#' e '?' podem ser colocados em qualquer lugar no arquivo de texto. Porém, somente os 5 anteriores podem ser ajustados como entradas ou saídas.

Exemplo:

Posição do LCD	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20

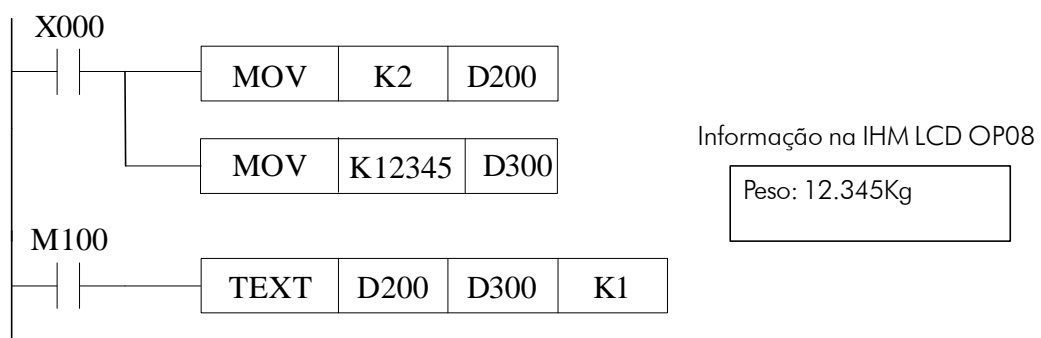
Arquivo de texto 1:

Registro D	2000		2001		2002		2003		2004		2005		2006		2007		2008		2009	
Conteúdo	C	o	m	p	r	i	m	:		#	#	.	#	#	#		c	m		

Arquivo de texto 2:

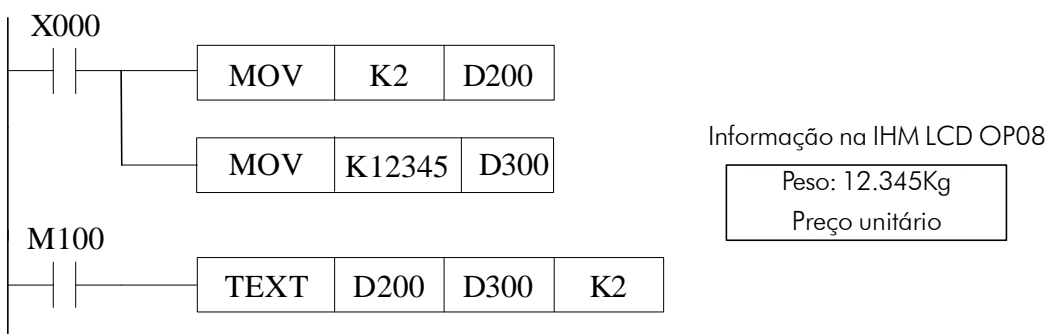
Registro D	2010		2011		2012		2013		2014		2015		2016		2017		2018		2019	
Conteúdo	P	e	s	o				:		#	#	.	#	#	#		k	g		

Arquivo de texto 3:

Registro D	2020		2021		2022		2023		2024		2025		2026		2027		2028		2029	
Conteúdo	P	r	e	ç	o		u	n	i	t.	:			$	?	?	?	?	?	

Descrição:

1, Quando X000 é setada, 2 será levado para D200 enquanto 12345 será levado para D300;

2, Quando M100 for setado, a instrução TEXT será habilitada. D8284 tem o valor padrão 13, D200 serão escritos em D8280; 300 a D8285. Então, OP07/08 entrará em modo F192.

3, F192 irá operar pela primeira vez. Enquanto D8280=D200=2, OP07/08 irá apresentar o arquivo 2 na primeira linha do display. Devido a um '#' no arquivo 2, 12345 em D300 será apresentado no lugar de '#'.

Descrição:

1) Quando X000 é setada, 2 será levado para D200 enquanto 12345 será levado para D300;

2) Quando M100 for setado, a instrução TEXT será habilitada. D8284 tem o valor padrão 13, D200 serão escritos em D8280; somatória dos dados em D200 e 1 será escrito em D8281, 300 a D8285, D8286. Então, OP08 entrará em modo F192.

3) F192 irá operar pela primeira vez. Enquanto D8280=D200=2, D8281=3, OP08 irá apresentar o arquivo 2 na primeira linha do display e o arquivo 3 na segunda linha. Além disso, 12345 em D300 será apresentado no lugar de '#' e os dados de entrada pelas teclas serão armazenados em D301.

3.17 COMPARAÇÕES LÓGICAS

Comparações lógicas - FNC 220 à FNC 249

Conteúdo:

LD□□ -	LD compare	FNC 224 à 230
E□ -	E compare	FNC 232 à 238
OU□ -	OU compare	FNC 240 à 246

Lista de símbolos:

D – Endereço de destino.

S – Endereço de origem.

m, n - Número de endereços ativos, bits ou uma constante operacional.

Sufixos numéricos adicionais serão anexados se houver mais de um operando com a mesma função, ex: D1, S3 ou para endereços de lista/tabelados D3+0, S+9 etc.

MSB – Bit mais significante, por vezes usado para indicar o sinal matemático de um número, ex: positivo = 0, e negativo = 1.

LSB – Bit menos significante.

Modificações de instrução:

□ □ □ - Instrução de 16 bits, onde □ □ □ identifica a instrução mnemônica.

□ □ □ P - Instrução de 16 bits habilitada por pulso (simples).

D□ □ □ - Instrução de 32 bits.

D□ □ □ P -Instrução de 32 bits habilitada por pulso (simples).

★ – Uma instrução repetitiva que mudará o valor de destino a cada varredura menos quando for habilitada por um pulso.

☆ – Um operando que não pode ser indexado, ex: A adição de V ou Z ou é inválida ou não terá efeito para o valor do operando.

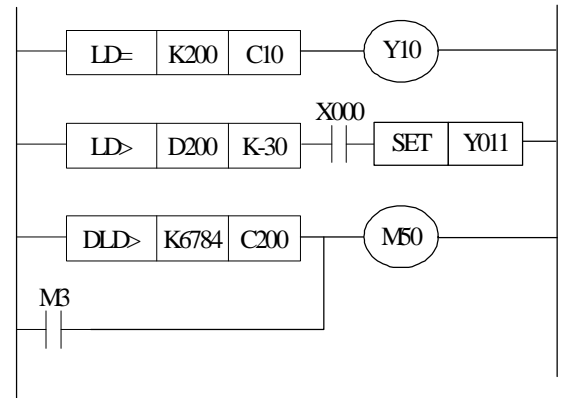
3.17.1 LD COMPARE (FNC 224 A 230)

Mnemônico	Função	Operandos		Passos do Programa
		S	D	
LD□ (Compare LoaD) onde□ é=, >, <, <>, □, □	Contato de comparação inicial. Ativo quando a comparação S1 □ S2 for verdadeira	K,H, KnX, KnY, KnM, KnS, T, C, D, V, Z		LD□: 5 passos DLD□: 9 passos

Operação:

Os valores de S1 e S2 são testados de acordo com a comparação da instrução. Se a comparação for verdadeira, então o contato LD está ativo. Se a comparação for falsa, então o contato LD não está ativo.

Pontos para observar:

As funções de comparação LD podem ser colocadas em qualquer lugar no programa que se possa colocar uma instrução padrão LD, por exemplo, sempre no início de um novo bloco.

Num F	16 bit	32 bit	Ativo quando	Inativo quando
224	LD=	DLD=	S1=S2	S1≠S2
225	LD>	DLD>	S1>S2	S1≤S2
226	LD<	DLD<	S1<S2	S1≥S2
228	LD<>	DLD<>	S1≠S2	S1=S2
229	LD≤	DLD≤	S1≤S2	S>1S2
230	LD≥	DLD≥	S1≥S2	S1<S2

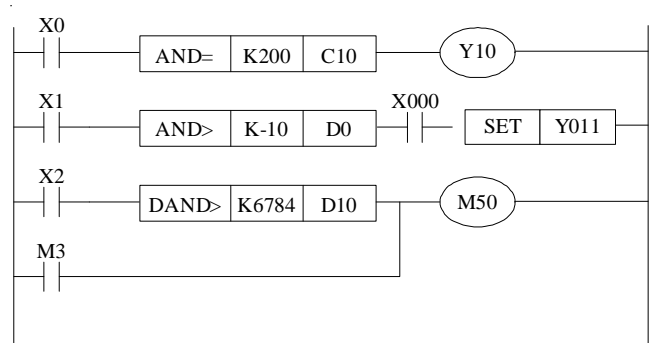
3.17.2 AND COMPARE (FNC 232 À 238)

Mnemônico	Função	Operandos		Passos do Programa
		S	D	
AND☐ (AND compare) onde☐ é =, >, <, <>, ≤ ≥	Contato de comparação serial. Ativo quando a comparação S1 ☐ S2 for verdadeira.	K,H, KnX, KnY, KnM, KnS, T, C, D, V, Z		E☐: 5 passos D E☐: 9 passos

Operação:

Os valores de S1 e S2 são testados de acordo com a comparação da instrução. Se a comparação for verdadeira, então o contato AND está ativo. Se a comparação for falsa, então o contato AND não está ativo.

Pontos para observar:

As funções de comparação AND podem ser colocadas em qualquer lugar no programa que se possa colocar uma instrução padrão AND, por exemplo, num contato de conexão serial.

Função	16 bits	32 bits	Ativo quando	Inativo quando
FNC 232	AND =	D AND =	S1=S2	S1≠S2
FNC 233	AND >	D AND >	S1>S2	S1≤S2
FNC 234	AND <	D AND <	S1<S2	S1≥S2
FNC 236	AND <>	D AND <>	S1≠S2	S1=S2
FNC 237	AND ≤	D AND ≤	S1≤S2	S>1S2
FNC 238	AND ≥	D AND ≥	S1≥S2	S1<S2

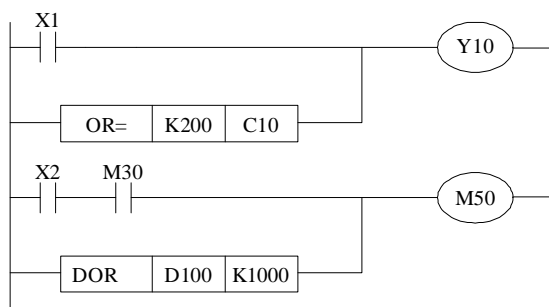
3.17.3 OR COMPARE (FNC 240 À 246)

Mnemônico	Função	Operandos		Passos do Programa
		S	D	
OR ☐ (OR compare) onde ☐ é =, >, <, <>, ≤, ≥	Contato de comparação paralelo. Ativo quando a comparação S1 ☐ S2 for verdadeira.	K,H, KnX, KnY, KnM, KnS, T, C, D, V, Z		OU☐: 5 passos DOU☐: 9 passos

Operação:

Os valores de S1 e S2 são testados de acordo com a comparação da instrução. Se a comparação for verdadeira, então o contato OR está ativo. Se a comparação for falsa, então o contato OR não está ativo.

Pontos para observar:

As funções de comparação OR podem ser colocadas em qualquer lugar no programa que se possa colocar uma instrução padrão OR, por exemplo, num contato de conexão paralela.

Função	16 bits	32 bits	Ativo quando	Inativo quando
FNC 240	OR=	D OR =	S1=S2	S1≠S2
FNC 241	OR >	D OR >	S1>S2	S1≤S2
FNC 242	OR <	D OR <	S1<S2	S1≥S2
FNC 244	OR <>	D OR <>	S1≠S2	S1=S2
FNC 245	OR ≤	D OR ≤	S1≤S2	S>1S2
FNC 246	OR ≥	D OR ≥	S1≥S2	S1<S2

CAPÍTULO 4

MARCADORES ESPECIAIS

4 MARCADORES ESPECIAIS

4.1 STATUS DO CLP – MARCADORES (M)

Marcador especial	Função	Operação
M8000	Status de RUN (Contato NA)	CLP em run M8061 M8000 M8001 M8002 M8003
M8001	Status de RUN (Contato NF)	
M8002	Pulso inicial (Contato NA)	
M8003	Pulso inicial (Contato NF)	
M8004	Erro	É setado quando uma ou mais marcadores M8060, 8061, M8063~8067 estiverem setados.
M8005	Carga baixa da bateria de bakc-up	Setado quando a tensão da bateria estiver abaixo do nível mínimo aceitável.
M8006	Memoriza a condição de carga baixa da bateria	Seta quando a tensão da bateria estiver muito baixa e reseta quando uma nova bateria for instalada.

4.2 ENDEREÇOS DO RELÓGIO (M)

Marcador epecial	Função	Operação
M8010		
M8011	Sinal cíclico de 10ms	5ms LIGADO/5ms DESLIGADO
M8012	Sinal cíclico de 100ms	50ms LIGADO/50ms DESLIGADO
M8013	Sinal cíclico de 1s	0,5s LIGADO/0,5s DESLIGADO
M8014	Sinal cíclico de 1min	30s LIGADO/30s DESLIGADO
M8015	Parada e ajuste do relógio	Pára a contagem do tempo e ativar modo de programação
M8016	Para de exibir o tempo do relógio	Pára de exibir o tempo do relógio
M8017	Offset de +/-30s	Offset de +/-30 s para o tempo interno
M8018	Detecção de RTC	Verifique se o RTC está habilitado.
M8019	Erro de RTC	Relógio está ajustado fora da faixa aceitável.

4.3 MARCADORES DE OPERAÇÃO (M)

Marcador especial	Função	Operação
M8020	Zero	Ligado quando o resultado da soma ou subtração for 0
M8021	Borrow	Ligado quando o resultado da subtração for menor que o menor numero negativo do sistema
M8022	Carry	Setado quando o resultado da soma deveria ser carregar
M8023		
M8024	Direção BMOV	(F15) 0: para frente,1:para trás
M8026	Modo RAMP	(F67)0: zera, 1: manter
M8027	Modo PR	（F77）0: 8bytes;1: 16bytes
M8029	Termina a execução das instruções	Na posição ON quando as instruções como DSW (F72) terminarem

4.4 STATUS DO CLP – MARCADORES (D)

Registrador especial	Função	Operação
D8001	TP03 tipo	0x
D8002	Versão	0x100 representa versão 1.00
D8003	Numero de Identificação	Somente leitura
D8004	Código de Erro	
D8005	Código de Alarme	
D8006	Capacidade do Programa	

4.5 RTC – RELÓGIO DE TEMPO REAL (D)

Registrador especial	Função	Operação
D8010	Apresentar Ciclo de varredura (0.1ms unidade)	
D8011	Menor ciclo de varredura	
D8012	Maior ciclo de varredura	
D8013	Segundo (0~59)	
D8014	Minuto (0~59)	
D8015	Hora	
D8016	Dia	
D8017	Mês	
D8018	Ano (00~99)	
D8019	Semana	

4.6 MODO DE OPERAÇÃO DO CLP - MARCADORES (M)

Marcador especial	Função	Operação
M8031	Zera todos os registradores não-retentivos (quando executando instrução END)	Quando M8031 sofre a transição de ON/OFF, o status Y/M/S/T/C e o valor atual de T/C/D são zerados. Porém, registradores de dados especiais não serão apagados.
M8032	Zera todos os registradores retentivos (quando executando instrução END)	Quando M8032 é setado, os registradores retentivos são zerados.
M8033	Mantém os dados dos registradores quando o CLP estiver em STOP. Ver Nota 1	Modo de armazenamento dos dados dos registradores 0: STOP→RUN, TP03 salva de acordo com a condição selecionada. 1: STOP→RUN,TP03 salva todos os dados
M8034	Desabilita as saídas	1: saída 0; 0: saída Y
M8035	Força modo de operação	
M8036	Força modo RUN	
M8037	Força modo STOP	
M8039	Modo de varredura constante	1: HABILITAR; 0: DESABILITAR Este registrador não pode ser alterado com o CLP energizado.

Nota 1:

Na seguinte tabela as colunas 'Endereço Inicial' e 'Endereço Final' podem ser modificadas dentro da faixa ajustada, definem o primeiro e o último marcador que terão seus status salvos pelo CLP. 0: salva conforme a condição programada. 1: salva todos os dados, independente da condição ou faixa programada.

Endereço	Mnemônico	Ponto	Início	Fim	Endereço Inicial	Endereço Final	Faixa de ajuste
Marcador	M	3072	0	3071	500	1023	0-1023
Seqüenciador	S	1000	0	999	500	999	0-999
Temporizador	T	256	0	255			
Contador de 16 bits	C	500	0	199	100	199	0-199
Contador de 32 bits	C	56	200	255	200	255	200-255
Registrador	D	8000	0	7999	200	511	0-511

4.7 MODO DO PC (D)

Registrador especial	Função	Operação
D8039	Ciclo de varredura constante	Default: 0, unidade: ms

4.8 MARCADORES PROGRAMAÇÃO LADDER (M)

Marcador especial	Função	Operação
M8040	STL desabilitar transferência	M8040 ON, a transferência STL está desabilitada.
M8041	STL iniciar transferência	Quando M8041 estiver setado, a transferência STL é habilitada em operação automática.
M8046	Estado do STL ON	Quando M8047 estiver setado e qualquer outro S0~S899 também, M8064 será energizado.
M8047	Habilitar monitor STL	Enquanto M8047 estiver setado, D8040~D8047 estarão desabilitados.
M8048	Anunciador ON	IVM8049 ON, e qualquer S900~S999 ON, M8048 estará setado.
M8049	Habilitar Anunciador	M8049 ON, D8049 está habilitado.

4.9 REGISTRADORES PROGRAMAÇÃO LADDER (D)

Registrador especial	Função	Operação
D8040	Mostra os 8 últimos estados ativos na faixa de S0 a S899, organizados em ordem crescente. (atualizados através da instrução END)	
D8041		
D8042		
D8043		
D8044		
D8045		
D8046		
D8047		
D8048		
D8049	O endereço mínimo para o estado de ON entre (S900 ~ S999)	

4.10 DESABILITAR INTERRUPÇÃO (M)

Marcador especial	Função	Operação
M8050	Desabilitar interrupção de entrada (I00x)	
M8051	Desabilitar interrupção de entrada (I10x)	
M8052	Desabilitar interrupção de entrada (I20x)	
M8053	Desabilitar interrupção de entrada (I30x)	
M8054	Desabilitar interrupção de entrada (I40x)	
M8055	Desabilitar interrupção de entrada (I50x)	
M8056	Desabilitar interrupção de tempo (I6xx)	
M8057	Desabilitar interrupção de tempo (I7xx)	
M8058	Desabilitar interrupção de tempo (I8xx)	
M8059	Desabilitar interrupção de contagem	I010~I060 desabilitar interrupção

4.11 MARCADOR DE AJUSTE DA CONTAGEM CRESCENTE/DECRESCENTE (M)

Marcador especial	Função	Operação
M8200	Ajuste da contagem crescente/decrescente para C200	
M8201	Ajuste da contagem crescente/decrescente para C201	
M8202	Ajuste da contagem crescente/decrescente para C202	
M8203	Ajuste da contagem crescente/decrescente para C203	
M8204	Ajuste da contagem crescente/decrescente para C204	
M8205	Ajuste da contagem crescente/decrescente para C205	
M8206	Ajuste da contagem crescente/decrescente para C206	
M8207	Ajuste da contagem crescente/decrescente para C207	
M8208	Ajuste da contagem crescente/decrescente para C208	
M8209	Ajuste da contagem crescente/decrescente para C209	
M8210	Ajuste da contagem crescente/decrescente para C210	
M8211	Ajuste da contagem crescente/decrescente para C211	
M8212	Ajuste da contagem crescente/decrescente para C212	
M8213	Ajuste da contagem crescente/decrescente para C213	
M8214	Ajuste da contagem crescente/decrescente para C214	
M8215	Ajuste da contagem crescente/decrescente para C215	
M8216	Ajuste da contagem crescente/decrescente para C216	
M8217	Ajuste da contagem crescente/decrescente para C217	
M8218	Ajuste da contagem crescente/decrescente para C218	
M8219	Ajuste da contagem crescente/decrescente para C219	
M8220	Ajuste da contagem crescente/decrescente para C220	
M8221	Ajuste da contagem crescente/decrescente para C221	
M8222	Ajuste da contagem crescente/decrescente para C222	
M8223	Ajuste da contagem crescente/decrescente para C223	
M8224	Ajuste da contagem crescente/decrescente para C224	
M8225	Ajuste da contagem crescente/decrescente para C225	
M8226	Ajuste da contagem crescente/decrescente para C226	
M8227	Ajuste da contagem crescente/decrescente para C227	
M8228	Ajuste da contagem crescente/decrescente para C228	
M8229	Ajuste da contagem crescente/decrescente para C229	
M8230	Ajuste da contagem crescente/decrescente para C230	
M8231	Ajuste da contagem crescente/decrescente para C231	
M8232	Ajuste da contagem crescente/decrescente para C232	
M8233	Ajuste da contagem crescente/decrescente para C233	
M8234	Ajuste da contagem crescente/decrescente para C234	
M8235	Ajuste da contagem crescente/decrescente para C235	
M8236	Ajuste da contagem crescente/decrescente para C236	
M8237	Ajuste da contagem crescente/decrescente para C237	
M8238	Ajuste da contagem crescente/decrescente para C238	
M8239	Ajuste da contagem crescente/decrescente para C239	
M8240	Ajuste da contagem crescente/decrescente para C240	
M8241	Ajuste da contagem crescente/decrescente para C241	
M8242	Ajuste da contagem crescente/decrescente para C242	
M8243	Ajuste da contagem crescente/decrescente para C243	
M8244	Ajuste da contagem crescente/decrescente para C244	
M8245	Ajuste da contagem crescente/decrescente para C245	
M8246	Monitor da contagem crescente/decrescente para C246	

Marcador especial	Função	Operação
M8247	Monitor da contagem crescente/decrescente para C247	
M8248	Monitor da contagem crescente/decrescente para C248	
M8249	Monitor da contagem crescente/decrescente para C249	
M8251	Monitor da contagem crescente/decrescente para C251	
M8252	Monitor da contagem crescente/decrescente para C252	
M8253	Monitor da contagem crescente/decrescente para C253	
M8254	Monitor da contagem crescente/decrescente para C254	

4.12 CAPACIDADE DO REGISTRO (D)

Registrador especial	Função	Operação
D8102	Conteúdo de registrador	

4.13 ENDEREÇOS (M)

Marcador especial	Função	Operação
M8061	Verificar hardware CLP	Erro no hardware CLP
M8064	Verificar parâmetro	
M8065	Verificar sintaxe	
M8066	Verificar programa	
M8067	Verificar operação	
M8068	Falha na operação de armazenamento de dados retentivos	
M8109	Verificar atualização de saída	
M8069	Verificar do barramento das entradas e saídas	

4.14 DETECÇÃO DE ERRO (D)

Registrador especial	Função	Operação
D8061	Código de erro	
D8063	Código de erro	
D8064	Código de erro	
D8065	Código de erro	
D8066	Código de erro	
D8067	Código de erro	
D8068	Código de erro	
D8109	Endereço de Y no erro de atualização da saída	

4.15 COMMUNICAÇÃO E LINK (M) I

Para porta RS485

Marcador especial	Função	Operação
M8121	RS485 porta de comunicação está pronta para enviar dados	RS, MBUS
M8122	Bit de envio de dados pela porta RS 485	RS, MBUS
M8123	Bit de finalização de recebimento de dados através da porta de comunicação RS485	RS, MBUS
M8124	Bit de erro na instrução de comunicação via porta RS 485 em MODBUS	MBUS
M8129	Bit de falha na transmissão de dados através da porta de comunicação RS485	RS, MBUS

Para porta de comunicação na unidade de expansão

Marcador especial	Função	Operação
M8321	Porta de comunicação no módulo de expansão está pronta para enviar dados	RS,MBUS
M8322	Bit de envio de dados pela porta de comunicação do módulo de expansão	RS,MBUS
M8323	Bit de finalização de recebimento de dados através da porta de comunicação do módulo de expansão	RS,MBUS
M8324	Bit de erro na instrução de comunicação via porta de comunicação do módulo de expansão em MODBUS	MBUS
M8329	Bit de falha na transmissão de dados através da porta de comunicação do módulo de expansão	RS,MBUS

Para RMIO

Marcador especial	Função	Operação
M8335	Dados RMIO em transmissão	
M8336	Erro na transmissão de dados RMIO (mestre)	
M8337	Erro na transmissão de dados RMIO (escravo 1)	
M8338	Erro na transmissão de dados RMIO (escravo 2)	
M8339	Erro na transmissão de dados RMIO (escravo 3)	
M8340	Erro na transmissão de dados RMIO (escravo 4)	
M8341	Porta de comunicação no módulo de expansão está programa para comunicação RMIO	
M8342	Porta de comunicação RS 485 está programada para comunicação RMIO	

4.16 COMUNICAÇÃO E LINK (D) I

Para porta RS485

Registrador especial	Função	Operação
D8120	Formato de comunicação	Porta de comunicação RS485 89Hex
D8121	Endereço	Somente leitura default: 01
D8122	Número de dados restantes a serem enviados pela porta RS485	
D8123	Número de dados recebidos RS485	
D8124	Caractere Inicial	Porta de comunicação RS485, RS instrução 02Hex
D8125	Caractere Final	Porta de comunicação RS485, RS instrução 03Hex
D8129	Tempo do watchdog da comunicação	Porta de comunicação RS485, RS e instrução MBUS

Para a porta de comunicação no módulo de expansão

Regisrtrador especial	Função	Operação
D8320	Formato de comunicação	Porta de comunicação no módulo de expansão (RS485/RS232）89Hex
D8321	Endereço	Porta de comunicação PC/PDA 89HEx
D8322	Numero de dados remanecentes de envio de dados	Porta de comunicação no módulo de expansão
D8323	Numero de dados recebidos RS485	Porta de comunicação no módulo de expansão
D8324	Caractere inicial	Porta de comunicação no módulo de expansão, RS instrução 02Hex
D8325	Caractere final	Porta de comunicação no módulo de expansão RS instrução 03Hex
D8329	Tempo do watchdog da comunicação	Porta de comunicação no módulo de expansão (RS e MBUS）

Para RMIO

Registrador especial	Função	Operação
D8373	Estado de ajuste do escravo RMIO	
D8374	Ajuste do escravo RMIO	
D8376	Escravo RMIO	
D8377	Ajuste do numero do escravo RMIO	
D8379	RMIO tempo para retransmissão	
D8380	Tempo de monitoração da rede RMIO	
D8331	Ciclo de varredura atual	
D8332	Ciclo de varredura máximo	
D8333	Contagem dos erros (mestre)	
D8334	Contagem dos erros (escravo 1)	
D8335	Contagem dos erros (escravo 2)	
D8336	Contagem dos erros (escravo 3)	
D8337	Contagem dos erros (escravo 4)	
D8338	Código de erro (mestre)	
D8339	Código de erro (escravo 1)	
D8340	Código de erro (escravo 2)	
D8341	Código de erro (escravo 3)	
D8342	Código de erro (escravo 4)	

4.17 COMUNICATION E LINK (M) II

DTLK

Marcador especial	Função	Operação
M8400	Erro no envio de dados (master)	
M8401	Erro no envio de dados (escravo 1)	
M8402	Erro no envio de dados (escravo 2)	
M8403	Erro no envio de dados (escravo 3)	
M8404	Erro no envio de dados (escravo 4)	
M8405	Erro no envio de dados (escravo 5)	
M8406	Erro no envio de dados (escravo 6)	
M8407	Erro no envio de dados (escravo 7)	
M8408	Erro no envio de dados (escravo 8)	
M8409	Erro no envio de dados (escravo 9)	
M8410	Erro no envio de dados (escravo 10)	
M8411	Erro no envio de dados (escravo 11)	
M8412	Erro no envio de dados (escravo 12)	
M8413	Erro no envio de dados (escravo 13)	
M8414	Erro no envio de dados (escravo 14)	
M8415	Erro no envio de dados (escravo 15)	
M8416	Envio de dados	
M8417	Porta de comunicação do módulo de expansão está ajustada para comunicação DTLK	
M8418	Porta RS485 está ajustada para comunicação DTLK	

4.18 COMUNICAÇÃO E LINK (D) II

DTLK

Registrador especial	Função	Operação
D8173	Ajustar estado do mestre	Link de dados
D8174	Ajustar estado do escravo	Link de dados
D8175	Ajustar estado do atualização dos dados	Link de dados
D8176	Ajustar endereço do Mestre	Link de dados
D8177	Ajustar endereço do escravo	Link de dados
D8178	Ajustar atualização dos dados	Link de dados
D8179	Tempo para retransmissão	Link de dados
D8180	Tempo de monitoração	Link de dados

Registrador especial	Função	Operação
D8401	Tempo atual do scan	
D8402	Ciclo de varredura máximo	
D8403	Número de erros no mestre	
D8404	Número de erros no escravo 1	
D8405	Número de erros no escravo 2	
D8406	Número de erros no escravo 3	
D8407	Número de erros no escravo 4	
D8408	Número de erros no escravo 5	
D8409	Número de erros no escravo 6	
D8410	Número de erros no escravo 7	
D8411	Número de erros no escravo 8	
D8412	Número de erros no escravo 9	
D8413	Número de erros no escravo 10	
D8414	Número de erros no escravo 11	
D8415	Número de erros no escravo 12	
D8416	Número de erros no escravo 13	
D8417	Número de erros no escravo 14	
D8418	Número de erros no escravo 15	
D8419	Código de erro (master)	
D8420	Código de erro (escravo 1)	
D8421	Código de erro (escravo 2)	
D8422	Código de erro (escravo 3)	
D8423	Código de erro (escravo 4)	
D8424	Código de erro (escravo 5)	
D8425	Código de erro (escravo 6)	
D8426	Código de erro (escravo 7)	
D8427	Código de erro (escravo 8)	
D8428	Código de erro (escravo 9)	
D8429	Código de erro (escravo 10)	
D8430	Código de erro (escravo 11)	
D8431	Código de erro (escravo 12)	
D8432	Código de erro (escravo 13)	
D8433	Código de erro (escravo 14)	
D8434	Código de erro (escravo 15)	

4.19 ALTA VELOCIDADE E POSIÇÃO (M)

Marcador especial	Função	Operação
M8130	Modo de comparação da zona de contagem de alta velocidade F55 (HSZ)	
M8131	Marcador de término para F55	
M8132	F55 (HSZ), F57 (PLSY) modo de velocidade	
M8133	F55, F57 marcador do término da execução	
M8134	Reservado	
M8135	Reservado	
M8136	Reservado	
M8137	Reservado	
M8138	Reservado	
M8139	Reservado	
M8140	FNC156 (ZRN) CLR habilitar saída do sinal	
M8141	Reservado	
M8142	Reservado	
M8143	Reservado	
M8144	Reservado	
M8145	Y000 paradas das saídas de pulso	
M8146	Y001 paradas nas saídas de pulso	
M8147	Y000 Monitoramento da saída de pulsos (ocupado/ler)	
M8148	Y001 Monitoramento da saída de pulsos (ocupado/ler)	
M8149	Reservado	

4.20 EXPANSÃO (M)

Marcador especial	Função	Operação
M8158	Reservado	
M8159	Reservado	
M8160	F17(XCH) SWAP	
M8161	Modo de processamento 8 octal (76,80,83,87,84)	
M8162	Modo de link paralelo de alta velocidade High speed parallel link mode	
M8163		
M8164		
M8165	Reservado	
M8166	Reservado	
M8167	F71(HKY)HEX processamento de dados	
M8168	F13(SMOV)DE HEX processamento	
M8169		

4.21 ALTA VELOCIDADE E POSIÇÃO (D)

Registrador especial	Função	Operação
D8130	Comparação da zona de contagem de alta velocidade High speed counter zone compare	
D8131	Contém o número do registro atual sendo processado na tabela de comparação HSZ quando a operação PLSY for habilitada	
D8132	Freqüência (HSZ, PLSY)	
D8133		
D8134	Pulso alvo	
D8135		
D8136	Valor acumulado do pulso de saída de Y000 e Y001	
D8137		
D8138		
D8139		
D8140	F57, 59 (PLSR), Valor acumulado para o pulso de saída de Y000 ou valor presente da instrução de posição	
D8141		
D8142	F57, 59 (PLSR), Valor acumulado para o pulso de saída de Y001 ou valor presente da instrução de posição	
D8143		
D8144		
D8145	Valor do Offset para F156, F158, F159	
D8146	Maior velocidade	
D8147		
D8148	Valor inicial	
D8149		

4.22 OP07/08 (M)

Marcador especial	Função	Operação
M8280	Tecla F1	
M8281	Tecla F2	
M8282	Tecla F3	
M8283	Tecla F4	
M8284	Tecla F5	
M8285	Tecla F6	
M8286	Tecla F7	
M8287	Tecla F8	
M8288	Tecla F9	
M8289	Tecla F10	
M8290	Tecla F11	
M8291	Tecla F12	
M8292	Para cima	
M8293	Para baixo	
M8294	Esquerda	
M8295	Direita	
M8296	Tecla TMR	
M8297	Tecla CNT	
M8298	Tecla ENT	
M8299	Tecla MOD1	
M8300	Tecla MOD2	
M8301	Tecla ESC	
M8302	Reservado	
M8303	Reservado	

4.23 OP07/08 (D)

Registrador especial	Função	Operação
D8280	Conteúdo da primeira linha defaulted	
D8281	Conteúdo da segunda linha defaulted	
D8282	Usuário do Conteúdo da primeira linha definido	
D8283	Usuário do Conteúdo da primeira linha definido	
D8284	OP07/08 Ajuste do modo de display	
D8285	OP07/08 Modo de display atual	
D8286	OP07/08 faixa do número do display	
D8287	Código de erro	
D8288		
D8289	Número atual para o timer mode	
D8290	Número atual para o modo de contador	
D8291	Número atual para o modo de usuário 1	
D8292	Número atual para o modo de usuário 2	
D8293	Número atual para o modo de usuário 3	
D8294	Número atual para o modo de usuário 4	
D8295	Conteúdo da primeira linha para o modo F192	
D8296	Conteúdo da segunda linha para o modo F192	
D8297	Ajuste do formato dos dados 1	
D8298	Ajuste do formato dos dados 2	
D8299	Ajuste do formato dos dados 3	
D8300	Ajuste do formato dos dados 4	

4.24 AD/DA (M)

Marcador especial	Função	Operação
M8257	Qualidade total dos módulos AD está errado	
M8258	Qualidade total do canal do módulo DA está errado	

4.25 AD/DA (D)

Registrador especial	Função	Operação
D8256	Número TP02-4AD	
D8257	Número TP03-AD （0~7）	
D8258	Canais TP02-2DA （0，2）	
D8259	Canais TP03-DA （0~8）	
D8260	Modo do filtro AD	
D8261	AD1～4 ajuste do modo de canal	
D8262	AD5～8 ajuste do modo de canal	
D8263	AD9～12 ajuste do modo de canal	
D8264	AD13～16 ajuste do modo de canal	
D8265	AD17～20 ajuste do modo de canal	
D8266	AD21～24 ajuste do modo de canal	
D8267	AD25～28 ajuste do modo de canal	
D8268	AD29～32 ajuste do modo de canal	
D8269	AD33～36 ajuste do modo de canal	
D8270	AD37～40 ajuste do modo de canal	
D8271	AD41～44 ajuste do modo de canal	
D8272	AD45～48 ajuste do modo de canal	
D8273	AD49～52 ajuste do modo de canal	
D8274	AD53～56 ajuste do modo de canal	
D8275	AD57～60 ajuste do modo de canal	
D8276	Reservado	
D8277	DA1～4 ajuste do modo de canal	
D8278	DA5～8 ajuste do modo de canal	
D8279	DA9～10 ajuste do modo de canal	
D8381	Dados do canal 1 DA	
D8382	Dados do canal 2 DA	
D8383	Dados do canal 3 DA	
D8384	Dados do canal 4 DA	
D8385	Dados do canal 5 DA	
D8386	Dados do canal 6 DA	
D8387	Dados do canal 7 DA	
D8388	Dados do canal 8 DA	
D8389	Dados do canal 9 DA	
D8390	Dados do canal 10 DA	
D8436	Dados do canal 1 AD	
D8437	Dados do canal 2 AD	
D8438	Dados do canal 3 AD	
D8439	Dados do canal 4 AD	
D8440	Dados do canal 5 AD	
D8441	Dados do canal 6 AD	
D8442	Dados do canal 7 AD	
D8443	Dados do canal 8 AD	
D8444	Dados do canal 9 AD	
D8445	Dados do canal 10 AD	
D8446	Dados do canal 11 AD	
D8447	Dados do canal 12 AD	
D8448	Dados do canal 13 AD	
D8449	Dados do canal 14 AD	
D8450	Dados do canal 15 AD	
D8451	Dados do canal 16 AD	
D8452	Dados do canal 17 AD	
D8453	Dados do canal 18 AD	
D8454	Dados do canal 19 AD	
D8455	Dados do canal 20 AD	
D8456	Dados do canal 21 AD	
D8457	Dados do canal 22 AD	
D8458	Dados do canal 23 AD	
D8459	Dados do canal 24 AD	
D8460	Dados do canal 25 AD	
D8461	Dados do canal 26 AD	
D8462	Dados do canal 27 AD	
D8463	Dados do canal 28 AD	

Registrador especial	Função	Operação
D8464	Dados do canal 29 AD	
D8465	Dados do canal 30 AD	
D8466	Dados do canal 31 AD	
D8467	Dados do canal 32 AD	
D8468	Dados do canal 33 AD	
D8469	Dados do canal 34 AD	
D8470	Dados do canal 35 AD	
D8471	Dados do canal 36 AD	
D8472	Dados do canal 37 AD	
D8473	Dados do canal 38 AD	
D8474	Dados do canal 39 AD	
D8475	Dados do canal 40 AD	
D8476	Dados do canal 41 AD	
D8477	Dados do canal 42 AD	
D8478	Dados do canal 43 AD	
D8479	Dados do canal 44 AD	
D8480	Dados do canal 45 AD	
D8481	Dados do canal 46 AD	
D8482	Dados do canal 47 AD	
D8483	Dados do canal 48 AD	
D8484	Dados do canal 49 AD	
D8485	Dados do canal 50 AD	
D8486	Dados do canal 51 AD	
D8487	Dados do canal 52 AD	
D8488	Dados do canal 53 AD	
D8489	Dados do canal 54 AD	
D8490	Dados do canal 55 AD	
D8491	Dados do canal 56 AD	
D8492	Dados do canal 57 AD	
D8493	Dados do canal 58 AD	
D8494	Dados do canal 59 AD	
D8495	Dados do canal 60 AD	

CAPÍTULO 5

TABELAS DE ENDEREÇO DO TPW-03

5 TABELAS DE ENDEREÇOS DO TPW-03

Item		Especificação	Comentários
Método de controle de operação		Operação cíclica por programa armazenado	
Método de controle I/O		Processamento cíclico do programa com atualização dos endereços quando o comando END é executado	O controlador possui instrução para forçar a atualização dos I/O's
Tempo de processamento de cada instrução		0,31 a 0,9 µs por instrução básica de programação	
Linguagem de programação		Ladder ou lista de instruções	
Capacidade do programa		8000 /16000 passo de programação	
Número de instruções		Instruções básicas: 36 Instruções especiais: 139	Máximo de 139 instruções especiais estão disponíveis
Configuração I/O		Número de pontos de I/O igual à 256, dependendo da seleção do usuário (Número máximo de entradas endereçáveis por software 256 e 256 saídas)	
Marcador auxiliar (Endereços M)	Geral	7680 pontos	M0 a M7679
	Especial	512 pontos	M8000 a M8511
Marcadores de estados (Endereços S)	Geral	4096 pontos	S0 a S4095
	Retentivos	500 pontos	S500 a S999
	Inicial	10 pontos	S0 a S9
	Anunciador	100 pontos	S900 a S999
Temporizador (T)	100 mseg	Faixa: 0 a 3.276,7 seg 200 pontos	T0 a T199
	10 mseg	Faixa: 0 a 327,67 seg 46 pontos	T200 a T245
	1 mseg retentivo	Faixa: 0 a 32,767 seg 4 pontos	T246 a T249
	100 mseg retentivo	Faixa: 0 a 3.276,7 seg 6 pontos	T250 a T255
Contadores (C)	Geral 16 bits	Faixa: 1 a 32.767 contagens 200 pontos	C0 a C199 Tipo: 16 bit unidirecional
	Retentivo 16 bits	100 pontos	C100 a C199 Tipo: 16 bit unidirecional
	Geral 32 bits	Faixa: -2.147.483.648 a 2.147.483.647 35 pontos	C200 a C234 Tipo: 32 bit bidirecional
	Retentivo 32 bits	15 pontos	C220 a C234 Tipo: 15 bit bidirecional
Contadores de alta velocidade (C)	1 canal	Faixa: -2,147,483,648 a +2,147,483,647 contagens Regra geral: Somente é possível a leitura de contadores com freqüência igual ou menor que 20kHz. Nota: todos os contadores são retentivos	C235 a C240 6 pontos
	1 canal com entrada de habilitação		C241 a C245 5 pontos
	2 canais		C246 a C250 5 pontos
	A/B canal		C251 a C255 5 pontos
Registradores de dados (D)	Geral	8000 pontos	D0 a D7999 Tipo: par de registros de armazenamento de dados de 16 bits para endereço de 32 bits
	Especial	512 pontos	Da faixa D8000 a D8511 Tipo: registro de armazenamento de dados de 16 bits
	Índice	16 pontos	V0 a V15 e Z0 a Z15 Tipo: registro de armazenamento de dados de 16 bits
Ponteiros (P)	Utilizando a instrução CALL	256 pontos	P0 a P255
	Utilizando interrupções	6 pontos de entrada, 3 temporizadores e 6 contadores.	I00□ a I50□ I6□□ a I8□□ I010 a I060
Níveis de aninhamento		8 pontos para usar com MC e MRC	N0 a N7
Números	Decimal K	16 bits: -32.768 a +32.767 32 bits: -2.147.483.648 a +2.147.483.647	
	Hexadecimal H	16 bit: 0000 a FFFF 32 bit: 00000000 a FFFFFFFF	

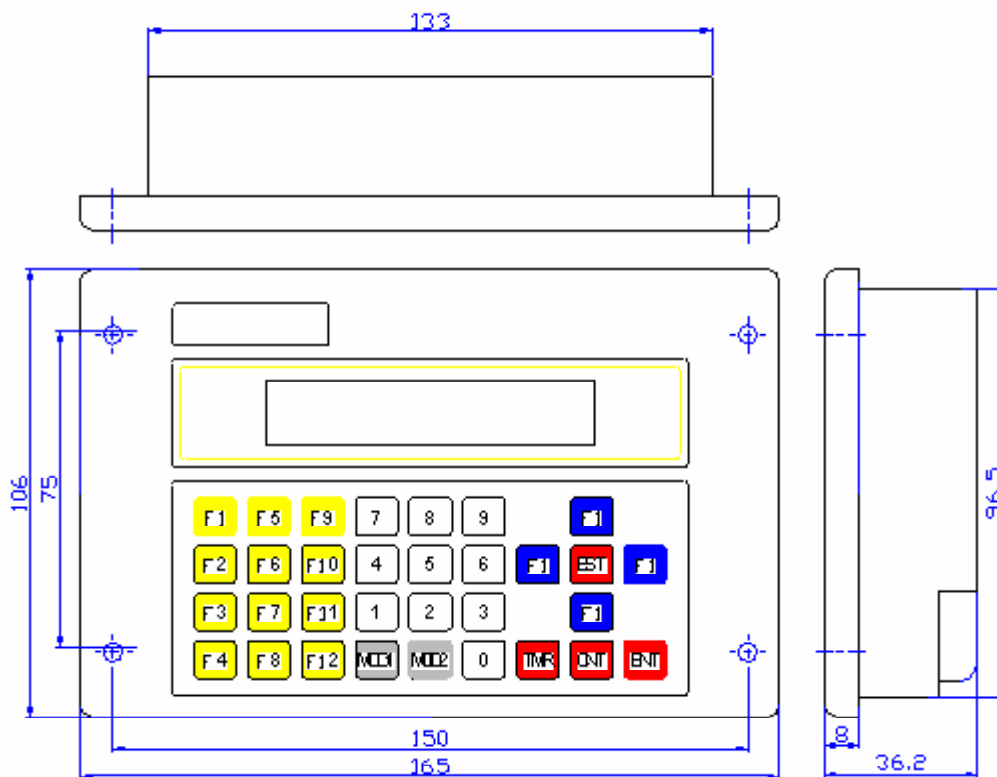
OP7/OP8 INSTRUÇÃO DE OPERAÇÃO

6 OP07/OP08 INSTRUÇÃO DE OPERAÇÃO

6.1. ESPECIFICAÇÕES ELÉTRICAS

Ítem	Especificação OP07	Especificação OP08
LCD	20 caracteres x 2 linhas，5x7 Pontos cada caractér，STN fonte, luz de fundo verde amarelado, 82(W)x18(H)mm	20 caracteres x 2 linhas，5x7 Pontos cada caractér，STN fonte, luz de fundo verde amarelado, 82(W)x18(H)mm
Teclado	32 teclas com membrana à prova de água	32 teclas com membrana à prova de água
Temperatura de Operação	5℃～50℃	5℃～50℃
Temperatura de Armazenagem	-10℃～60℃	-10℃～60℃
Humidade da operação e armazenagem	40％～60％RH（não condensado），30％～85％	40％～60％RH（não condensado），30％～85％
Ambiente de operação	Sem gotas condensadas nem gás corrosivo	Sem gotas condensadas nem gás corrosivo
Alimentação	5V fornecido pelo TPW-03	Necessita alimentação externa 24V DC
Consumo de energia	5V：200mA	24V：80mA
Dimensão	Comprimento：165mm Largura：106mm altura：62mm	Comprimento：165mm Largura：106mm altura：62mm
Peso	245g	260g
Porta de comunicação	RS422 PG cabo de comunicação	RS422 PG cabo de comunicação, Ou TR+/TR- conexão
Parafuso de montagem	M4*5（4）	M4*5（4）
Acessório	TP-302MC，instrução de instalação	instrução de instalação

6.2 CONFIGURAÇÃO DIMENCIONAL E CONEXÕES

(1) Dimensões (mm)

(2) Fiação OP08

Existem tres métodos de fiação para o OP08. O primeiro é conectar o TR+/TR- aos terminais TP03 RS485 A/B com o cabo trançado. O segundo é conectar ao TP302MC. O último é conectar ao TP305MC.

Os tres métodos de fiação devem ser fonecidos com alimentação 24V DC.

6.3 FUNÇÕES DAS TECLAS

Função F1~F12: Habilitar o relé correspondente por um ciclo de varredura.

Tecla de Dígito: Entrada de dígito sob o estado previamente ajustado.

Função TMR: Entra no modo de ajuste do TEMPORIZADOR.

Função CNT: Entra no modo de ajuste do CONTADOR.

Função ENT: Entrada com os dados.

Função ESC: Cancela os dados antes de entrar com eles. Sob o modo de display de erro, modo de ensino ou modo de display F192, volta ao modo de display inicial.

Função da tecla esquerda-direita: Esquerdo e direito movimenta o cursor sob o estado previamente ajustado.

Função para cima-para baixo: Modifica o código, sob modo de display F192, movimenta o cursor.

Função MOD1, MOD2: Função definida pelo usuário.

Quando uma tecla de função for pressionada, o relé interno correspondente é habilitado (ON por um ciclo de varredura), e abaixo está a tabela de informações da tecla de função.

Teclas	Relé interno
F1	M8280
F2	M8281
F3	M8282
F4	M8283
F5	M8284
F6	M8285
F7	M8286
F8	M8287

Teclas	Relé interno
F9	M8288
F10	M8289
F11	M8290
F12	M8291
Para cima	M8292
Para baixo	M8293
Esquerda	M8294
Direita	M8295

Teclas	Relé interno
TMR	M8296
CNT	M8297
ENT	M8298
MOD1	M8299
MOD2	M8300
ESC	M8301
reservado	M8302
reservado	M8303

Sob o modo de display de erro, todas as teclas de função correspondente ao RELÉ ficam desabilitadas, menos a tecla ESC.

Sob modo Ensino, ações do RELÉ correspondentes a TMR, CNT, MOD1, MOD2 não podem ser habilitas o tempo todo. Ações do RELÉ correspondentes às teclas F1~F12 não podem ser habiliadas até que a tecla ENT seja pressionada.

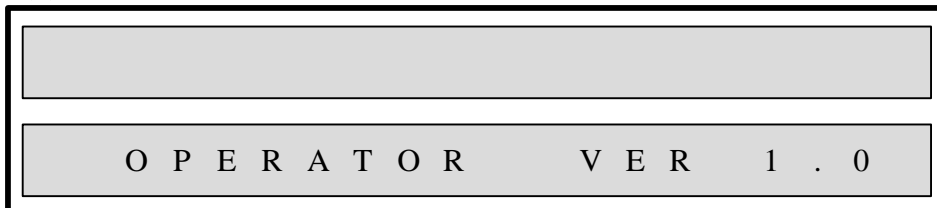
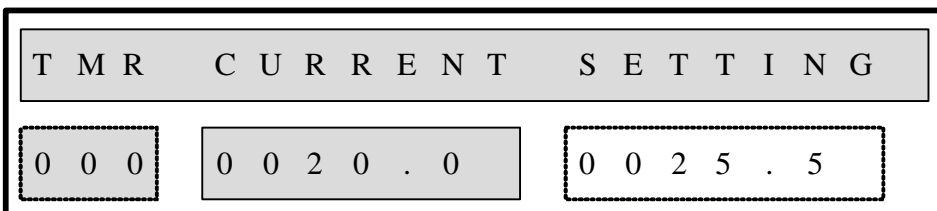
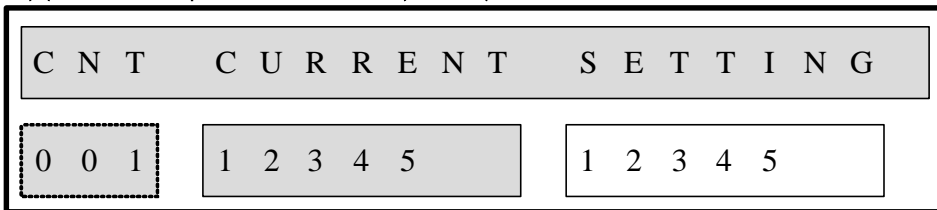
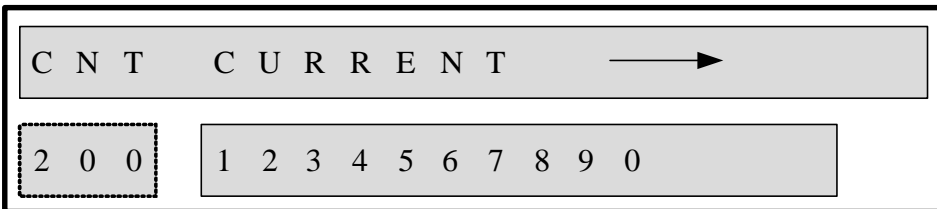
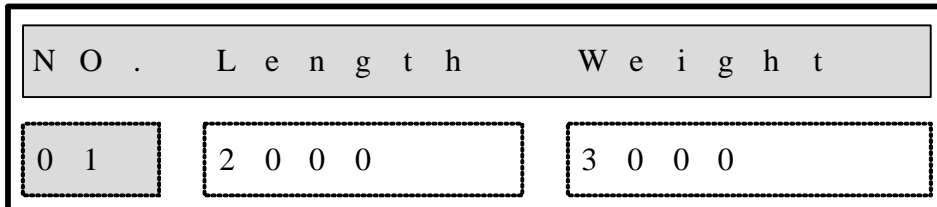
7.4 USO E EXEMPLOS DA OP07/08

Lista de registros do OP07/08

Fução e modo do display LCD	Registro do sistema							Registro de dados	
	Ajuste do modo	Entrada do modo	Define o número de registro do arquivo		Registro para ajustar o Nº	Registro para a faixa limite do número	Outro registro		
	D8284	D8285	LCD Primeira linha	LCD Segunda linha				Código	número
Modo de display inicial	1	1	D8282	D8283					
Modo de ajuste									
Modo de ajuste do TIMER	2	2			D8289	D8286		T0-T511 D3300-D3811	512 512
Modo de ajuste do CONTADOR(16bit)	3	3			D8290	D8286		C0-C199 D3812-D4011	200 200
Modo de ajuste do CONTADOR(32bit)	3	3			D8290	D8286		C200-C255 D4012-D4123	56 112
Modo do usuário 1	4	4	D8280		D8291	D8286	D8297, D8298, D8300	D4124-D4163	40
Modo do usuário 2	5	5	D8280		D8292	D8286	D8297, D8298, D8299, D8300	D4164-D4223	60
Modo do usuário 3	6	6			D8293	D8286	D8297	D4224-D4243	20
Modo do usuário 4	7	7	D8280		D8294	D8286		M1600-M1615	16
Modo de display									
Modo de display 1	8	8	D8280				D8297, D8298, D8299	D4244-D4246	3
Modo de display 2	9	9	D8280				D8297, D8298	D4247,D4248	2
Modo de display 3	14	14	D8280					D4271	1
Modo de display 4	15	15					D8297, D8298, D8299, D8300	D4272-D4275	4
Modo de display do histórico de erros	10	10		D8287				D4250-D4254	5
Modo de display de erro	11	11		D8287				D4249	1
Modo de ensino	12	12		D8281		D8286		D4255-D4270	16
Modo de display F192	13	13	D8280	D8281			D8295, D8296		

Nota: Após executar o comando F192, o TPW-03 irá escrever 13 para o registro D8284.

Tela LCD

(1) Modo de Display Inicial

(2) Modo de Ajuste do TIMER

(3) Modo de Ajuste do Contador (16 bits)

(3) Modo de Ajuste do Contador (32 bits)

(4) Modo do Usuário 1

(5) Modo do Usuário 2

(6) Modo do Usuário 3

(7) Modo do Usuário 4

(8) Modo de Display 1

(9) Modo de Display 2

(10) Modo de Display do Histórico de Erros

(11) Modo de Display de Erros

Registro do sistema utilizado no OP07/08

Registro para o LCD

D8280: O valor em D8280 é o número do arquivo (File), que será visualizado na 1ª linha, com esceção do modo de display inicial. O valor de D8280 deve ser mantido entre 1 e 130.

D8281: O valor em D8281 é o número do arquivo (File), que será visualizado na 2ª linha, com exceção do modo de display inicial. O valor de 8281 deve ser mantido entre 1 e 130.

D8282: O valor deste registro é o número do arquivo (File) que será visualizado na 1ª linha do LCD no modo de display inicial.

D8283: O valor deste registro é o número do arquivo (File) que será visualizado na 1ª linha do LCD no modo de display inicial.

Ajuste do modo e confirmação do registro

D8284: Para ajustar modo atual do LCD. Em TPW-03 ajuste o modo LCD, OP07/08 irá mudar para o modo de display correspondente após ler o valor do registro.

D8285: Para ajustar modo atual do LCD. OP07/08 lê o registro D8284 e muda para o modo. O valor de registro será escrito para D8285.

Faixa de limite do N° (número)

D8286: Para exibir a faixa do n° (0 ~15/19/255/511, e n° 15/19/255/511 se o número for maior que 15/19/255/511.)

Ajustar o N° (número)

D8289: Para exibir o n° do 'modo de ajuste do TEMPORIZADOR' após o número de display ser trocado, OP07/08 irá ler D8289 do TPW-03 e o valor será visualizado na tela (D8289=1 →N° 1,D8289 = 2 → N° 2 ...), Pressione a tecla para cima/ para baixo para ajustar o valor do N°, então o valor será escrito para D8289 em TPW-03 pelo OP07/08.

D8290: Para exibir o n° do 'modo de ajuste do CONTADOR'. O processo de ajuste é o mesmo para D8289.

D8291: Para exibir o n° do 'modo do usuário 1'... . O processo de ajuste é o mesmo para D8289.

D8292: Para exibir o n° do 'modo do usuário 2'... . O processo de ajuste é o mesmo para D8289.

D8293: Para exibir o n° do 'modo do usuário 3'... . O processo de ajuste é o mesmo para D8289.

D8294: Para exibir o n° do 'modo do usuário 4'... . O processo de ajuste é o mesmo para D8289.

Modo F192

D8295: ajuste do sistema - Sob modo F192, a primeira linha exibe número de registro '#####'. O número de registro mais 1 é o número de registro para "?????". (Por exemplo, D8295i×100, o número de registro para "#####" é D0100, e o número de registro para "?????" é D0101.)

D8296: ajuste do sistema - Sob modo F192, a segunda linha exibe número de registro '#####'. O número de registro mais 1 é o número de registro para "?????". (Por exemplo, D8295i×100, o número de registro para "#####" é D0100, e o número de registro para "?????" é D0101.)

O sistema principal irá automaticamente ajustar os valores nos dois registros acima conforme a operação da instrução F192-TEXT

Modo de display de erro

D8287: O endereço de início do arquivo ERRO adiciona o CÓDIGO DE ERRO entre D4250-D4254. O arquivo the ERRO atual será usado.
(Fórmula: (D8287) + (CÓDIGO DE ERRO) - 1 = Número do arquivo).

Registro para o ajuste do modo de display

D8297 define o formato do valor atual 1 sob o modo do usuário e o modo de display.

O byte alto exibe o número de dígitos (faixa: 1~5, outro número será por padrão o 5).

O byte baixo exibe o número de dígitos decimais, menor que o número de dígitos (faixa: 0~4, outro número será por padrão o 4).

Se D8297=0502H, a primeira figura irá exibir 5 dígitos e 2 dígitos decimais.

D8298 define o formato do valor previamente ajustado 2 (ajuste é igual ao D8297).

D8299 define o formato do valor previamente ajustado 3 (ajuste igual ao D8297).

D8300 define o formato do valor previamente ajustado 1 sob o modo do usuário 1/2 e modo de display 4: (quando D8300=1 exibe o sinal '+', D8300=2 exibe o sinal '-' outros não exibem nenhum sinal) ;

Tela LCD

(LCD: 20 Caracteres x 2 linhas)

Posição	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
Primeira linha																				
Segunda linha			O	P	E	R	A	T	O	R			V	E	R		1	.	0	

Arquivo(File) e registro digital D**** / posição do display de LCD

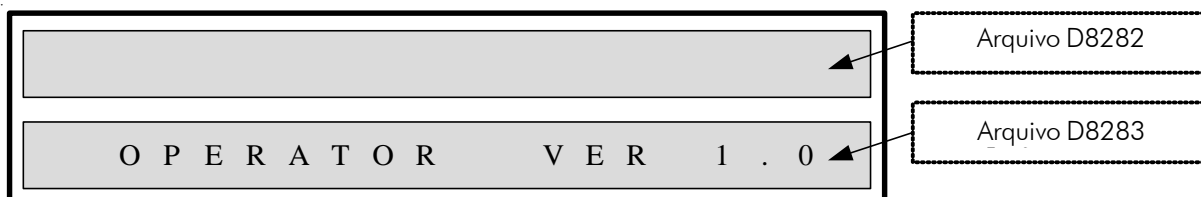
Posição do LCD	01	02	03	04	05	06	07 - 14	15	16	17	18	19	20
File 1	D2000		D2001		D2002		D2003-D2006	D2007		D2008		D2009	
File 2	D2010		D2011		D2012		D2013-D2016	D2017		D2018		D2019	
File 3	D2020		D2021		D2022		D2023-D2026	D2027		D2028		D2029	
File 4 ~ File 128	D2030~D3279 （125 Files *10 Registros digitais); Todo File armazenará 20 caracteres ASCII compartilhando 10 registros digitais D.												
File 129	D3280		D3281		D3282		D3283-D3286	D3287		D3288		D3289	
File 130	D3290		D3291		D3292		D3293-D3296	D3297		D3298		D3299	

(1) Modo de Display Inicial

Quando a força estiver ON, ou quando, sob outro modo, D8284 for ajustado em 1 sob outros modos, ou a tecla ESC for pressionada sob o modo de display de erro ou modo F192, OP07/08 entra no modo de display inicial.

Após mudar para o modo de display inicial, 1 será escrito para D8285.

Tela de display LCD

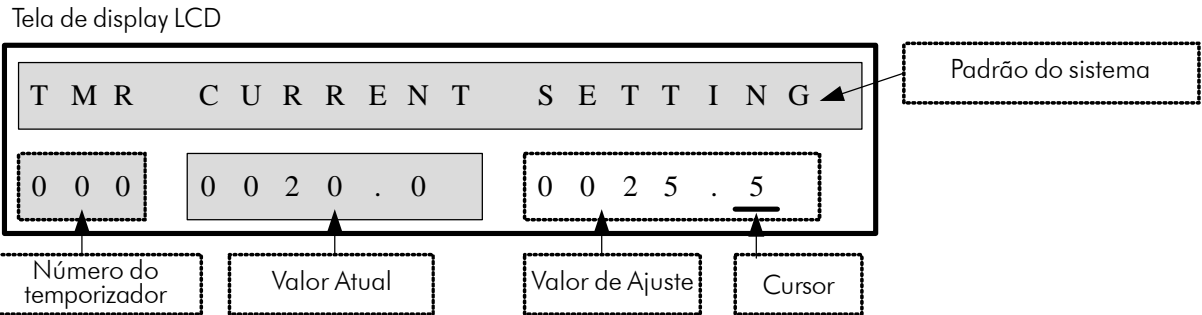
Se o valor em D8282 e D 8283 for 0, a tela LCD exibe:

Posição	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
Primeira linha																				
Segunda linha			O	P	E	R	A	T	O	R			V	E	R		1	.	0	

Primeira linha do LCD: exibe arquivo D8282. Se o arquivo=0 ou maior que 130, a primeira linha será em branco.

Segunda linha do LCD: exibe arquivo D8283. Se o arquivo=0 ou maior que 130, a segunda linha será texto por padrão

(2) Modo de ajuste do temporizador

Quando, sob outro modo, D8284 é programado para 2 ou pressionando a tecla TMR, a tela irá entrar no modo de ajuste do TEMPORIZADOR.

Será escrito 2 para D8285 após a tela ser mudada. (Nota: 0P07/08 também escreverá 2 para D8284 quando pressionar a tecla TMR para entrar no modo de ajuste do Temporizador).

Posição	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
Primeira linha	T	M	R		C	U	R	R	E	N	T		S	E	T	T	I	N	G	
Segunda linha	0	2	0			6	5	5	3	.	5			6	5	5	3	.	5	

Nota:

A unidade do valor atual e do ajustado é o segundo.O ponto de decimal é padrão, e aparece de acordo com diferentes tipos de temporizador.

Número dos temporizadores válidos no TPW-03: T0~T511.

T0~T199, T250~T255 são temporizadores de 100ms com um dígito decimal.

T200~T245 são temporizadores de 10ms com 2 dígitos decimais.

T246~T249, T256~T511 são temporizadores de 1ms com 3 dígitos decimais.

D8286 define o número máximo.

D8289 define o número de início para o modo de ajuste do TEMPORIZADOR.

Teclas direita e esquerda podem movimentar o cursor.

Informação de registro:

Número do TMR	Valor de D8289	Número do Cronômetro para valor atual	Número do registro para o valor ajustado
0	0	T0 valor atual	D3300
1	1	T1 valor atual	D3301
2	2	T2 valor atual	D3302
...	...	...	...
254	154	T254 valor atual	D3554
255	255	T255 valor atual	D3555

Número do TMR	Valor de D8289	Número do Cronômetro para valor atual	Número do registro para o valor ajustado
256	256	T256 valor atual	D3556
257	257	T257 valor atual	D3557
...	...	...	...
...	...	...	...
510	510	T510 valor atual	D3810
511	511	T511 valor atual	D3811

Nota 1:

OP07/08 irá automaticamente exibir o valor do registro e escrever os principais dados de entrada para o registro correspondente.

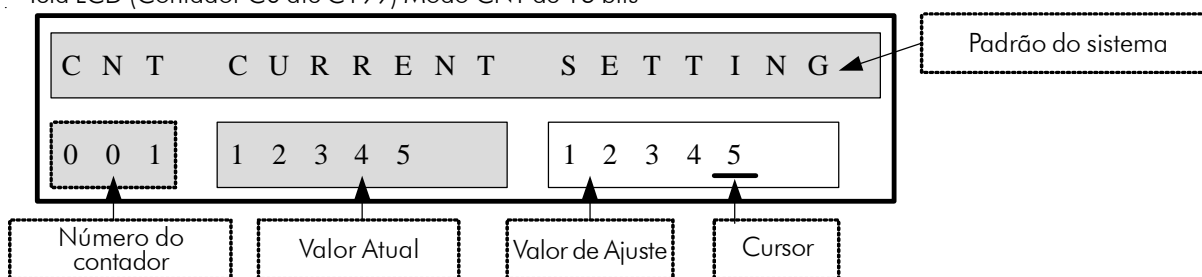
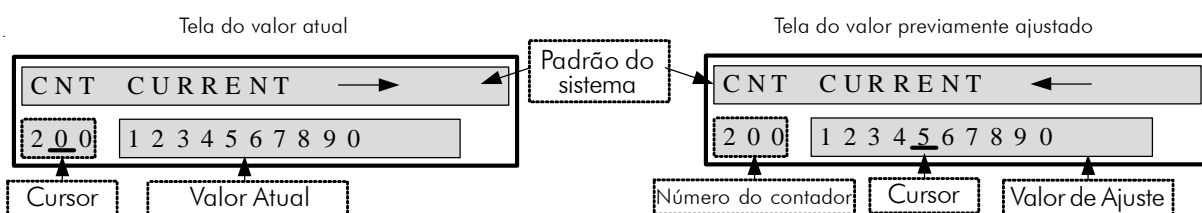
Nota 2:

TPW-03 tem 512 temporizadores (T0 ~T511), OP07/08 usa os registros de dados D3300 ~ D3811 (total de 512) para armazenar o valor atual do temporizador. Consequentemente, existem no máximo 512 cronômetros disponíveis.

(3) Modo de ajuste do contador (Contador C0 até C255)

Quando, sob outro modo, D8284 é programado para 3 ou pressionando a tecla CNT, a tela irá entrar no modo de ajuste do CONTADOR.

Será escrito 3 para D8285 após a tela ser mudada para o modo de ajuste do contador. (Nota: OP07/08 também irá escrever 3 para D8284 quando pressionar a tecla CNT para entrar no modo de ajuste do Contador).

Tela LCD (Contador C0 até C199) Modo CNT de 16 bits

Tela LCD (Contador C200 até C255) Modo CNT de 32 bits

Posição	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
Primeira linha	C	N	T		C	U	R	R	E	N	T		S	E	T	T	I	N	G	
Segunda linha	0	2	0			6	5	5	3	5				6	5	5	3	5		

D8286 define o número máximo.

D8290 define o número inicial para o modo de ajuste CONTADOR.

Teclas direita e esquerda podem mover o cursor ou trocar a tela de display para modo de CONTADOR de 32 bits.

Informação de registro:

Número do CNT de 16 bit	D8290	Número do Contador para valor atual	Número do registro para o valor ajustado
0	0	C0 valor atual	D3812
1	1	C1 valor atual	D3813
2	2	C2 valor atual	D3814
...	...	...	...
198	154	C198 valor atual	D4010
199	255	C199 valor atual	D4011

Número do CNT de 16 bit	D8290	Número do Contador para valor atual	Número do registro para o valor ajustado
200	200	C200 valor atual	D4013 D4012
201	201	C201 valor atual	D4015 D4014
...	...	...	...
...	...	...	...
254	254	C254 valor atual	D4121 D4120
255	255	C255 valor atual	D4123 D4122

Nota 1:

OP07/08 irá automaticamente exibir o valor de registro e escrever os principais dados de entrada para o registro correspondente.

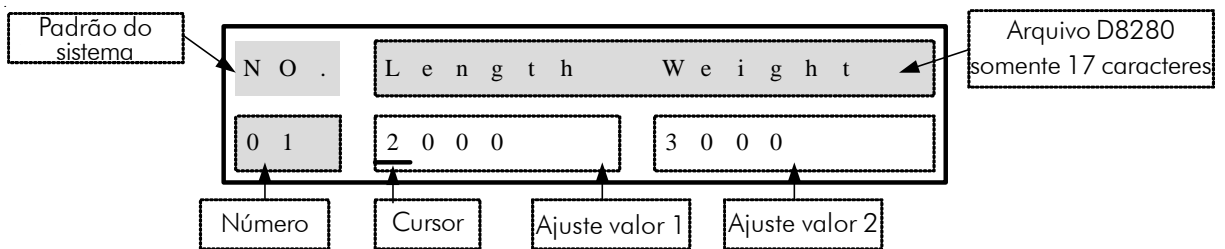
Nota 2:

TPW-03 tem 200 contadores (C0~C199), OP07/08 usa D3812~D4011 registros de dados (total 200) para armazenar o valor atual do cronômetro compartilhando 200 registros de dados.

Nota 3:

TPW-03 tem 56 contadores (C200~C255), OP07/08 usa D4012~D4123 registros de dados (total 56) para armazenar o valor previamente ajustado do cronômetro compartilhando 112 registros de dados.

(4) Modo do usuário (1- 19)

Quando,sob outro modo, D8284 é programado para 4, a tela entrará no modo de usuário 1.
Será escrito 4 para D8285 depois que a tela mudar para este modo.

Posição	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
Primeira linha	N	R			L	e	n	g	h	t			W	e	i	g	h	t		
Segunda linha	2	0				6	5	5	3	5			6	5	5	3	5			

O conteúdo da primeira linha do LCD corresponderá a um arquivo (File). Por exemplo, D8280=1, o LCD mostrará os primeiros 17 caracteres do arquivo 1 (File 1).

Posição do LCD	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20			
File 1		L	e	n	g	h	t			W	e	i	g	h	t					
	D2000		D2001		D2002		D2003		D2004		D2005		D2006		D2007		D2008 Byte Alto	D2008 Byte Baixo		

D8286: define o número máximo.

D8291: define o número de início para o modo do usuário 1.

D8280: define o número do arquivo (File) que pode mostrar 17 caracteres.

D8297: define o formato do valor 1 pré-programdo:

Byte alto mostra número de dígitos (faixa: 1~5, outros números serão por padrão 5).

Byte baixo mostra número de dígitos decimais menor que o número de dígitos (faixa: 0~4, outros números serão por padrão 4).

Se D8297=0502H, o primeiro número mostrará 5 dígitos e 2 dígitos decimais.

D8298 define o formato do valor 2 pré-programado (ajuste é igual ao D8297).

D8300 define o formato do valor 3 pré-programado (1 mostra o sinal '+', 2 mostra o sinal '-' outros mostram nenhum sinal);

As teclas direita e esquerda podem mover o cursor

Registro de informações (valor 1 e valor 2 pré-programado)

NR	Valor D8291	Valor 1 pré-programado	Valor 2 pré-programado
0	0	D4124	D4125
1	1	D4126	D4127
2	2	D4128	D4129
3	3	D4130	D4131
4	4	D4132	D4133
5	5	D4134	D4135
6	6	D4136	D4137
7	7	D4138	D4139
8	8	D4140	D4141
9	9	D4142	D4143

NR	Valor D8291	Valor 1 pré-programado	Valor 2 pré-programado
10	10	D4144	D4145
11	11	D4146	D4147
12	12	D4148	D4149
13	13	D4150	D4151
14	14	D4152	D4153
15	15	D4154	D4155
16	16	D4156	D4157
17	17	D4158	D4159
18	18	D4160	D4161
19	19	D4162	D4163

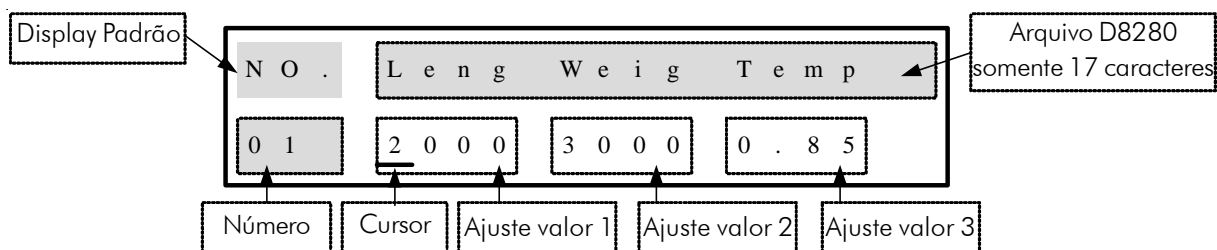
Nota: O LCD OP07/08 automaticamente mostrará os dados lidos dos registros correspondentes e escreverá os dados para registros relevantes.

Há até 20 grupos (Nr.0~Nr.19) disponíveis para o modo de usuário 1.

(5) Modo de usuário 2 (0 - 19)

Quando, sob outro modo, D8284 é programado para 5, a tela entrará no modo de usuário 2.

Será escrito 5 para D8285 depois que a tela mudar para este modo.

Tela LCD

Posição	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
Primeira linha	N	O		L	e	n	g			W	e	i	g			T	e	m	p	
Segunda linha	2	0		6	5	5	3	5		6	5	5	3	5		6	5	5	3	5

O conteúdo da primeira linha do LCD corresponderá a um arquivo (File). Por exemplo, D8280=1, o LCD mostrará os primeiros 17 caracteres do arquivo 1 (File 1).

Posição do LCD	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	
File 1	L	e	n	g			W	e	i	g			T	e	m	p		
	D2000		D2001		D2002		D2003		D2004		D2005		D2006		D2007		D2008 Byte Alto	D2008 Byte Baixo

D8286: define o número máximo.

D8292: define o número de início para o modo do usuário 1.

D8280: define o número do arquivo que pode mostrar 17 caracteres.

D8297: define o formato do valor 1 pré-programado:

Byte alto mostra número de dígitos (faixa: 1~5, outros números serão por padrão 5).

Byte baixo mostra número de dígitos decimais menor que o número de dígitos (faixa: 0~4, outros números serão por padrão 4).

Se D8297=0502H, o primeiro número mostrará 5 dígitos e 2 dígitos decimais.

D8298: define o formato do valor 2 pré-programado (ajuste é igual ao D8297).

D8300: define o formato do valor 3 pré-programado:(1 mostra o sinal '+', 2 mostra o sinal '-' outros mostram nenhum sinal);

As teclas direita e esquerda podem mover o cursor.

Registro de informações (ajuste do valor 1, ajuste do valor 2, ajuste do valor 3):

Nr Grupo	D8292	Valor 1	Valor 2	Valor 3
0	0	D4164	D4165	D4166
1	1	D4167	D4168	D4169
2	2	D4170	D4171	D4172
3	3	D4173	D4174	D4175
4	4	D4176	D4177	D4178
5	5	D4179	D4180	D4181
6	6	D4182	D4183	D4184
7	7	D4185	D4186	D4187
8	8	D4188	D4189	D4190
9	9	D4191	D4192	D4193

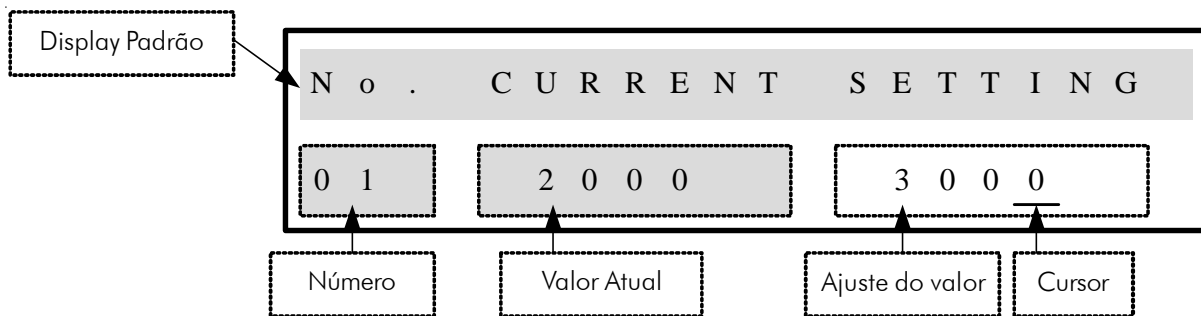
Nr Grupo	D8292	Valor 1	Valor 2	Valor 3
10	10	D4194	D4195	D4196
11	11	D4197	D4198	D4199
12	12	D4200	D4201	D4202
13	13	D4203	D4204	D4205
14	14	D4206	D4207	D4208
15	15	D4209	D4210	D4211
16	16	D4212	D4213	D4214
17	17	D4215	D4216	D4217
18	18	D4218	D4219	D4220
19	19	D4211	D4222	D4223

Nota: O LCD OP07/08 automaticamente mostrará os dados lidos dos registros correspondentes e escreverá os dados para registros relevantes.

Há até 20 (Nr.0~Nr.19) grupos disponíveis para o modo de usuário 2.

(6) Modo do usuário 3 (0 - 19)

Quando,sob outro modo, D8284 é programado para 6, a tela entrará no modo de usuário 3.
Será escrito 6 para D8285 depois que a tela mudar para este modo.

Tela LCD

Posição	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
Primeira linha	N	O			C	U	R	R	E	N	T		S	E	T	T	I	N	G	
Segunda linha	2	0				6	5	5	3	.	5			6	5	5	3	.	5	

D8286: define o número máximo.
D8293: define o número de início/partida para o modo do usuário 3.
D8297: define o formato do valor 1 pré-programado:
Byte alto mostra número de dígitos (faixa: 1~5, outros números serão por padrão 5).
Byte baixo mostra número de dígitos decimais menor que o número de dígitos (faixa: 0~4, outros números serão por padrão 4).

Registro de informações (valor atual e valor de ajuste):

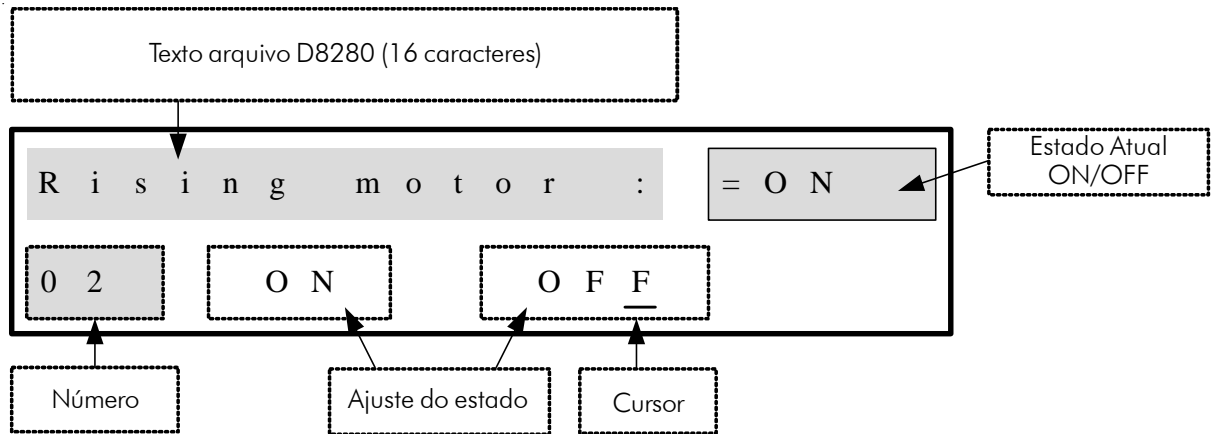
Nr Grupo	D8293	Valor atual	Valor de ajuste
0	0	D4224	D4224
1	1	D4225	D4225
2	2	D4226	D4226
3	3	D4227	D4227
4	4	D4228	D4228
5	5	D4229	D4229
6	6	D4230	D4230
7	7	D4231	D4231
8	8	D4232	D4232
9	9	D4233	D4233

Nr Grupo	D8293	Valor atual	Valor de ajuste
10	10	D4234	D4234
11	11	D4235	D4235
12	12	D4236	D4236
13	13	D4237	D4237
14	14	D4238	D4238
15	15	D4239	D4239
16	16	D4240	D4240
17	17	D4241	D4241
18	18	D4242	D4242
19	19	D4243	D4243

Nota: LCD OP07/08 automaticamente mostrará os dados lidos dos registros correspondentes e escreverá os dados para registros relevantes.
Há até 20 grupos (Nr.0~Nr.19) disponíveis para o modo de usuário 3.

(7) Modo do usuário 4 (0-15)

Quando, sob outro modo, D8284 é programado para 7, a tela entrará no modo de usuário 4.
Será escrito 7 para D8285 depois que a tela mudar para este modo.

Tela LCD

Posição	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
Primeira linha	R	i	s	i	n	g		m	o	t	o	r		:			=		O	N
Segunda linha	0	1		O	N			O	F	F										

O conteúdo da primeira linha do LCD corresponderá a um arquivo (File). Por exemplo, D8280+No.=1, o LCD mostrará os primeiros 16 caracteres do arquivo 1 (File 1).

Posição do LCD	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
File 1	R	i	s	i	n	g		m	o	t	o	r		:		
	D2000		D2001		D2002		D2003		D2004		D2005		D2006		D2007	

D8286: define o número máximo.
D8294: define o número de início para o modo do usuário 4.

Informações da bobina (status on/off):

Nr Grupo	D8294	Bobina (status on/off)
0	0	M1600
1	1	M1601
2	2	M1602
3	3	M1603
4	4	M1604
5	5	M1605
6	6	M1606
7	7	M1607

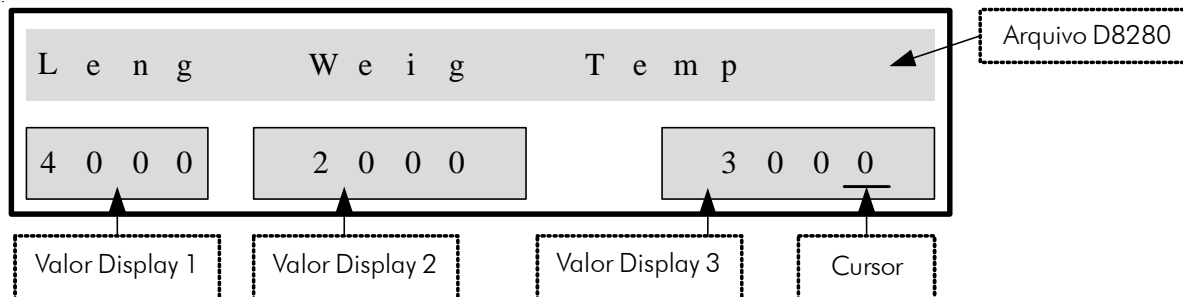
Nr Grupo	D8294	Bobina (status on/off)
8	8	M1608
9	9	M1609
10	10	M1610
11	11	M1611
12	12	M1612
13	13	M1613
14	14	M1614
15	15	M1615

Nota: O LCD OP07/08 automaticamente mostrará os dados lidos dos registros correspondentes e escreverá os dados para registros relevantes.

Há até 16 grupos (Nr.0~Nr.15) disponíveis para o modo de usuário 4.

(8) Modo *display* 1

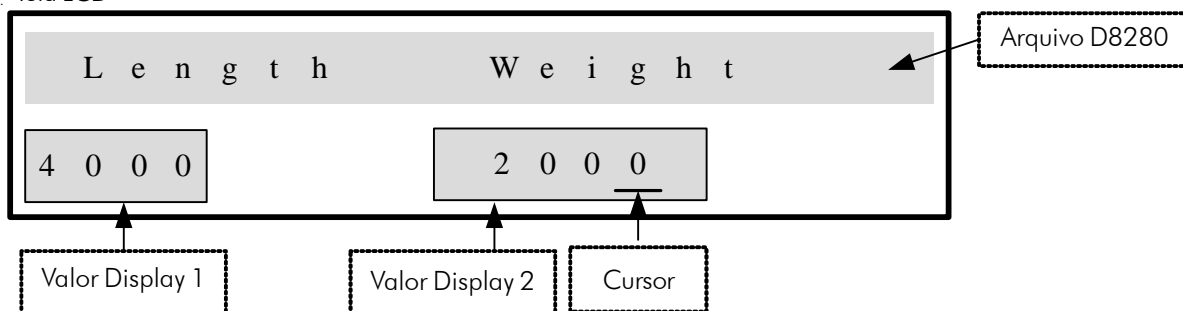
Quando, sob outro modo, D8284 é programado para 8, a tela entrará no modo 1.
Será escrito 8 para D8285 depois que a tela mudar para este modo.

Tela LCD

D8280: define o número do arquivo (File), que mostrará 20 caracteres.
D8291: define o número de início para o modo do usuário 1.
D8297: define o formato do valor 1 pré-programado:
Byte alto mostra número de dígitos (faixa: 1~5, outros números serão por padrão 5).
Byte baixo mostra número de dígitos decimais menor que o número de dígitos (faixa: 0~4, outros números serão por padrão 4).
Se D8297=0502H, o primeiro número mostrará 5 dígitos e 2 dígitos decimais.
D8298: define o formato do valor 2 pré-programado (ajuste é igual ao D8297).
D8299: define o formato do valor 3 pré-programado (ajuste é igual ao D8297).
D8300: define o formato do valor 3 pré-programado:(1 mostra o sinal '+', 2 mostra o sinal '-' outros mostram nenhum sinal).

Registro de informações:

Valor do *display* 1	Valor do *display* 2	Valor do *display* 3
D4244	D4245	D4246

(9) Modo *display* 2

Quando, sob outro modo, D8284 é programado para 9, a tela entrará no modo 2.
Será escrito 9 para D8285 depois que a tela mudar para este modo.

Tela LCD

D8280: define o número do arquivo (File), que mostrará 20 caracteres.

D8297: define o formato do valor 1 pré-programado:

Byte alto mostra número de dígitos (faixa: 1~5, outros números serão por padrão 5).

Byte baixo mostra número de dígitos decimais menor que o número de dígitos (faixa: 0~4, outros números serão por padrão 4).

Se D8297=0502H,o primeiro número mostrará 5 dígitos e 2 dígitos decimais.

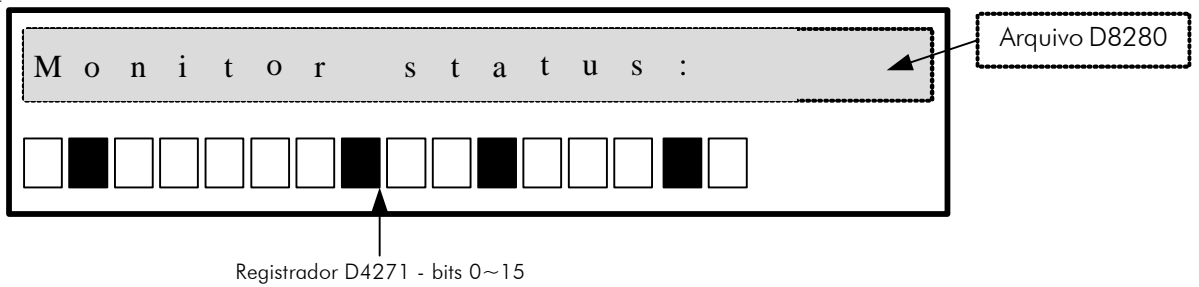
D8297: define o formato do valor 2 pré-programado (ajuste é igual ao D8297).

Registro de informações:

Valor do *display* 1	Valor do *display* 2
D4247	D4248

(10) Modo *display* 3

Quando, sob outro modo, D8284 é programado para 14, a tela entrará no modo 3.
Será escrito 14 para D8285 depois que a tela mudar para este modo.

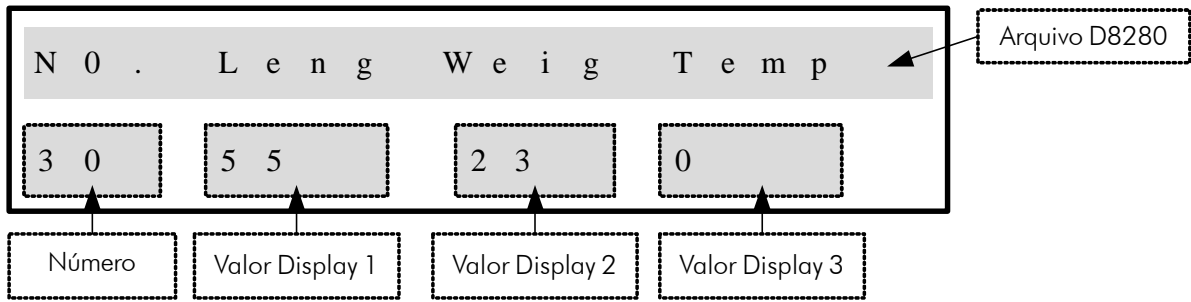
Tela LCD

D8280: define o número do arquivo (File), que mostrará 20 caracteres.

Registro de informações:

Registro do *display*
D4271

Nota: O LCD OP07/08 automaticamente mostrará o status lido do D4271. O Bit 15 será mostrado à esquerda. Quando este bit é igual a 1, o LCD mostrará o sinal ■, quando for 0, mostrará □.

(11) Modo *display* 4

Quando, sob outro modo, D8284 é programado para 15, a tela entrará no modo 4.
Será escrito 15 para D8285 depois que a tela mudar para este modo.

Tela LCD

D8280: define o número do arquivo (File),que mostrará 20 caracteres.

D8297: define o formato do valor 1 pré-programado:

Byte alto mostra número de dígitos (faixa: 1~5, outros números serão por padrão 5).

Byte baixo mostra número de dígitos decimais menor que o número de dígitos (faixa: 0~4, outros números serão por padrão 4).

Se D8297=0502H, o primeiro número mostrará 5 dígitos e 2 dígitos decimais.

D8298: define o formato do valor 2 pré-programado (ajuste é igual ao D8297).

D8299: define o formato do valor 3 pré-programado (ajuste é igual ao D8297).

D8300: define o formato do valor 1 pré-programado no modo de usuário 1/2 e no modo *display* 4: (quando D8300=1 mostra o sinal '+', 2 mostra o sinal '-' outros mostram nenhum sinal),

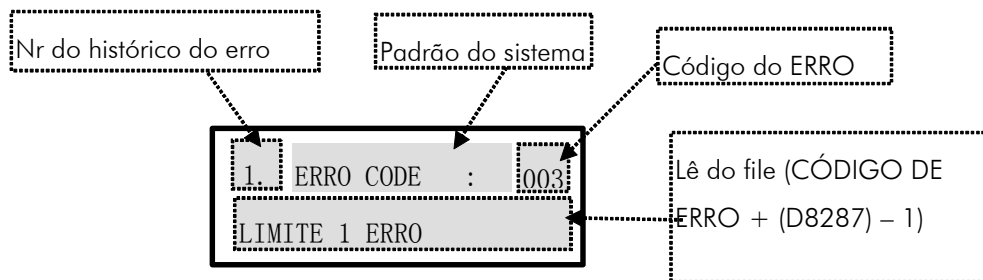
Registro de informações:

Número do *display*	Valor do *display* 1	Valor do *display* 2	Valor do *display* 3
D4272	D4273	D4274	D4275

Nota: O LCD OP07/08 automaticamente mostrará os dois dígitos mais baixos como número do display e mostrará os dados lidos dos D4273,D4374,D4275 nas posições 1, 2 e 3.

(12) Modo de display do histórico de erros (mostrar os últimos cinco erros diferentes)

Quando, sob outro modo, D8284 é programado para 10, a tela entrará no modo de *display* do histórico de erros.

Será escrito 10 para D8285 depois que a tela mudar para este modo.

Tela LCD

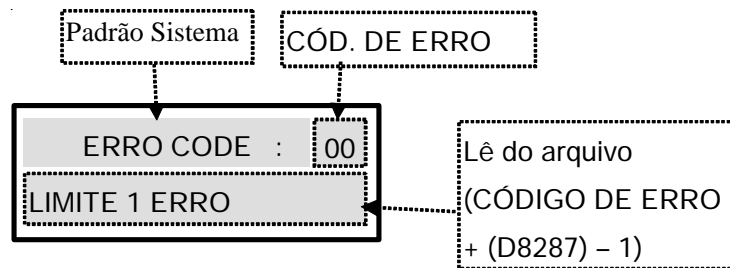
Posição	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
Primeira linha	1	.		C	Ó	D	I	G	O		D	E		E	R	R	O	:	3	
Segunda linha	L	I	M	I	T	E		1		E	R	R	O							

Para mostrar os últimos cinco erros diferentes;
O D8287: define o arquivo (File) de inicio do erro.
As teclas Up e Down estão disponíveis para ver outros erros.

Registro de informações:

No.	Registro para salvar o CÓDIGO DE ERRO	
1	D4250	Último erro
2	D4251	
3	D4252	
4	D4253	
5	D4254	

(13) Modo de *display* de erros

Quando, sob outro modo, D8284 é programado para 11, a tela entrará no modo de *display* de erros. Será escrito 11 para D8285 depois que a tela mudar para este modo.

Tela LCD

Posição	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
Primeira linha	1	.		C	Ó	D	I	G	O		D	E		E	R	R	O	:	3	
Segunda linha	L	I	M	I	T	E		1		E	R	R	O							

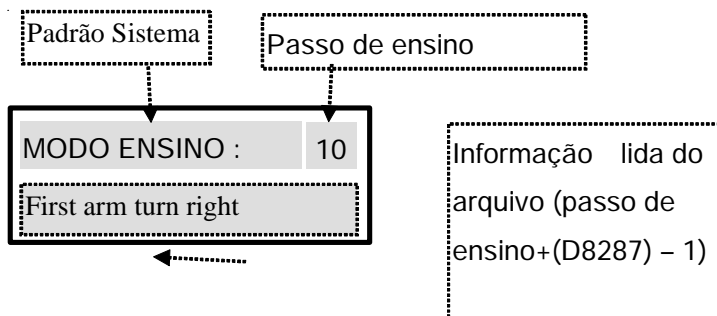
D8287: define o arquivo (file) de inicio do erro.
Entrando neste modo o registro de histórico de erros será atualizado automaticamente.
Pressionar a tecla ESC para sair para a tela inicial ou modificar o D8284 para outro modo.

Registror de informações:

Registro para salvar o código de erro
D4249

(14) Modo de display de ensino (1 grupo: 16 passos)

Quando, sob outro modo, D8284 é programado para 12, a tela entrará no modo de *display* de ensino.
Será escrito 12 para D8285 depois que a tela mudar para este modo.

Tela LCD

Posição	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
Primeira linha	M	O	D	O		E	N	S	I	N	O	:	P	A	S	S	O	=	1	0
Segunda linha	F	i	r	s	t		a	r	m		t	u	r	n		r	i	g	h	t

D8281: define o arquivo (File) de inicio para a instrução de ensino.

Somente de 1～12 estão disponíveis para os registros de ensino, e quanto a outros dígitos, o LCD estará em branco.

Pressionar a tecla ESC para sair para a tela inicial ou modificar o D8284 para outro modo.

F1 – F12 define o dígito:

Tecla função	dígito
F1	1
F2	2
F3	3
F4	4
F5	5
F6	6

Tecla função	dígito
F7	7
F8	8
F9	9
F10	10
F11	11
F12	12

Registro de informações:

Passo de ensino nr	Registro Correspondente
0	D4255
1	D4256
2	D4257
3	D4258
4	D4259
5	D4260

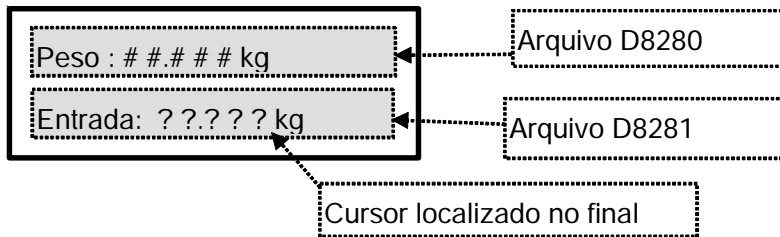
Passo de ensino nr	Registro Correspondente
6	D4261
7	D4262
8	D4263
9	D4264
10	D4265
11	D4266

Passo de ensino nr	Registro Correspondente
12	D4267
13	D4268
14	D4269
15	D4270

(15) Modo de display F192

Quando, sob outro modo, D8284 é programado para 13, a tela entrará no modo de *display* F192.
Será escrito 13 para D8285 depois que a tela mudar para este modo.

Este modo deverá cooperar com a instrução F192-NEXT.

Tela LCD

Depois que o F192 for habilitado, o OP07/08 salavrá o número do arquivo (File) para ser mostrado no LCD em D8280, D8281(D8280 é para o LCD Primeira linha, enquanto que o D8281 é para a Segunda linha), e também salvará os dados mostrados no D8295,D8296.

'#' e '?' podem ser colocados em qualquer lugar no arquivo, mas somente os 5 primeiros estarão disponíveis para entrada e saída.

No lugar de '#' na Primeira linha mostrará os dados no registro D8295, enquanto na Segunda linha, # mostrará os dados no D8296.

É possível entrar com dígitos no lugar de '?' no LCD. Os dados de entrada para a Primeira linha serão salvos no registro de dados (número =1 + dígito no D8295).

Pressionar a tecla ESC para sair para a tela inicial ou modificar D8284 para outro modo.